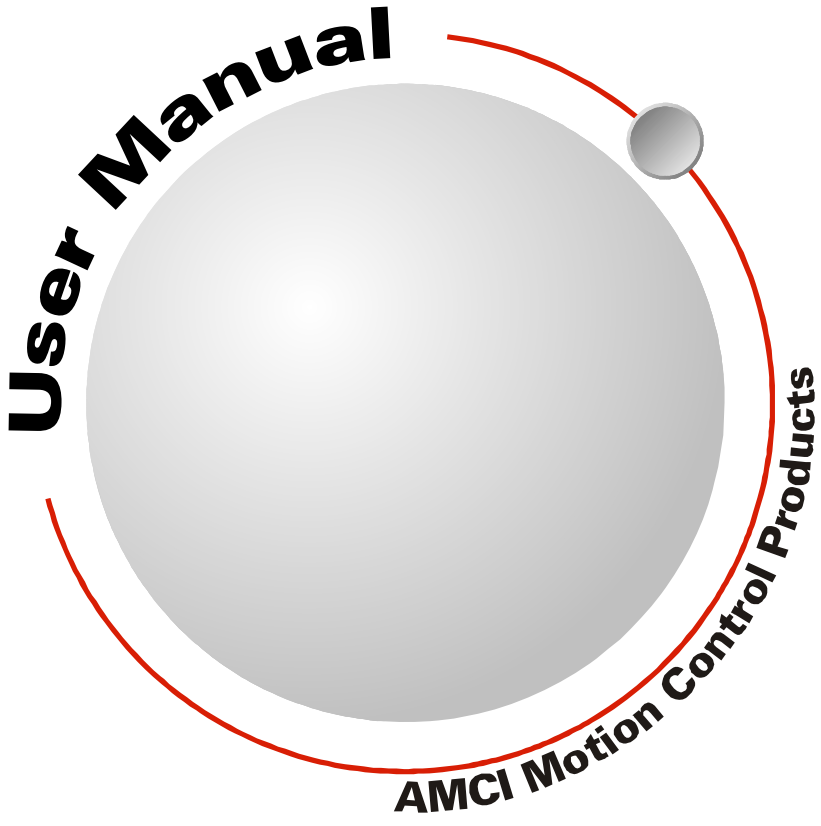
ADVANCED MICRO CONTROLS INC.

Manual #: 940-0S134

SMD23E and SMD24E Integrated Stepper Indexer/Driver/Motor

SMD23E-130
SMD23E-240
SMD23E-130E
SMD23E-240E
SMD23E-130-M12
SMD23E-240-M12
SMD23E-130E-M12
SMD23E-240E-M12
SMD23E-130-M12P
SMD23E-240-M12P
SMD23E-130E-M12P
SMD23E-240E-M12P
SMD24E-350
SMD24E-350E
SMD24E-350-M12
SMD24E-350E-M12
SMD24E-350-M12P
SMD24E-350E-M12P



EtherNet/IP™

Modbus

GENERAL INFORMATION

Important User Information

The products and application data described in this manual are useful in a wide variety of different applications. Therefore, the user and others responsible for applying these products described herein are responsible for determining the acceptability for each application. While efforts have been made to provide accurate information within this manual, AMCI assumes no responsibility for the application or the completeness of the information contained herein.

UNDER NO CIRCUMSTANCES WILL ADVANCED MICRO CONTROLS, INC. BE RESPONSIBLE OR LIABLE FOR ANY DAMAGES OR LOSSES, INCLUDING INDIRECT OR CONSEQUENTIAL DAMAGES OR LOSSES, ARISING FROM THE USE OF ANY INFORMATION CONTAINED WITHIN THIS MANUAL, OR THE USE OF ANY PRODUCTS OR SERVICES REFERENCED HEREIN.

No patent liability is assumed by AMCI, with respect to use of information, circuits, equipment, or software described in this manual.

The information contained within this manual is subject to change without notice.

This manual is copyright 2014 by Advanced Micro Controls Inc. You may reproduce this manual, in whole or in part, for your personal use, provided that this copyright notice is included. You may distribute copies of this complete manual in electronic format provided that they are unaltered from the version posted by Advanced Micro Controls Inc. on our official website: *www.amci.com*. You may incorporate portions of this documents in other literature for your own personal use provided that you include the notice “Portions of this document copyright 2014 by Advanced Micro Controls Inc.” You may not alter the contents of this document or charge a fee for reproducing or distributing it.

Standard Warranty

ADVANCED MICRO CONTROLS, INC. warrants that all equipment manufactured by it will be free from defects, under normal use, in materials and workmanship for a period of [18] months. Within this warranty period, AMCI shall, at its option, repair or replace, free of charge, any equipment covered by this warranty which is returned, shipping charges prepaid, within eighteen months from date of invoice, and which upon examination proves to be defective in material or workmanship and not caused by accident, misuse, neglect, alteration, improper installation or improper testing.

The provisions of the "STANDARD WARRANTY" are the sole obligations of AMCI and excludes all other warranties expressed or implied. In no event shall AMCI be liable for incidental or consequential damages or for delay in performance of this warranty.

Returns Policy

All equipment being returned to AMCI for repair or replacement, regardless of warranty status, must have a Return Merchandise Authorization number issued by AMCI. Call (860) 585-1254 with the model number and serial number (if applicable) along with a description of the problem during regular business hours, Monday through Friday, 8AM - 5PM Eastern. An "RMA" number will be issued. Equipment must be shipped to AMCI with transportation charges prepaid. Title and risk of loss or damage remains with the customer until shipment is received by AMCI.

24 Hour Technical Support Number

24 Hour technical support is available on this product. If you have internet access, start at www.amci.com. Product documentation and FAQ’s are available on the site that answer most common questions.

If you require additional technical support, call (860) 583-7271. Your call will be answered by the factory during regular business hours, Monday through Friday, 8AM - 5PM Eastern. During non-business hours an automated system will ask you to enter the telephone number you can be reached at. Please remember to include your area code. The system will page an engineer on call. Please have your product model number and a description of the problem ready before you call.

We Want Your Feedback

Manuals at AMCI are constantly evolving entities. Your questions and comments on this manual are both welcomed and necessary if this manual is to be improved. Please direct all comments to: Technical Documentation, AMCI, 20 Gear Drive, Terryville CT 06786, or fax us at (860) 584-1973. You can also e-mail your questions and comments to *techsupport@amci.com*

Table of Contents

General Information

Important User Information 2
Standard Warranty 2
Returns Policy 2
24 Hour Technical Support Number 2
We Want Your Feedback 2

About this Manual

Audience 7
Applicable Units 7
Revision Record 8
Revision History 8
Navigating this Manual 8
Manual Conventions 8
Trademark Notices 8
Where To Go From Here 9

Chapter 1: Introduction to the SMD23/4E

The SMD23/4E Family 11
General Functionality 12
Encoder Functionality 12
Indexer Functionality 13
Driver Functionality 14
Available Discrete Inputs 14
Home Input 14
CW Limit Switch or
CCW Limit Switch 14
Start Indexer Move Input 15
Emergency Stop Input 15
Stop Jog or Registration
Move Input 15
Capture Encoder Position Input 15
General Purpose Input 15
Status LED's 16
Module Status (MS) LED 16
Network Status (NS) LED 16
SMD23/4E Connectors 17
Input Connector 17
Ethernet Port 17
Link Status LED 17

Chapter 1: Introduction to the SMD23/4E (continued)

SMD23/4E-M12 Connectors 17
Input Connector 17
Ethernet Connector 18
Compatible Connectors
and Cordsets 18
Connectors 18
Ethernet Cordsets 19
Power Supply Sizing 19
Specifications 20
Torque and Power Curves 21

Chapter 2: Move Profiles

Units of Measure 23
Definition of Count Direction 23
Definition of Home Position 23
Definition of Starting Speed 23
Definition of Target Position 24
Relative Coordinates 24
Absolute Coordinates 24
Definition of Acceleration Types 24
Linear Acceleration 24
Triangular S-Curve Acceleration .. 25
Trapezoidal S-Curve Acceleration 25
A Simple Move 26
Controlled and Immediate Stops 27
Host Control 27
Hardware Control 27
Basic Move Types 27
Relative Move 27
Controlled Stop 28
Immediate Stop 28
Absolute Move 28
Controlled Stop 29
Immediate Stop 29
±Jog Move 29
Controlled Stop 30
Immediate Stop 30
±Registration Move 31
Controlled Stop 32
Immediate Stop 32
Encoder Moves 32
Controlled Stop 33
Immediate Stop 33

Chapter 2: Move Profiles (continued)

Assembled Moves 33
Blend Moves 33
Controlled Stop 34
Immediate Stop 34
Dwell Move 35
Controlled Stop 35
Immediate Stop..... 35
Assembled Move Programming 36
Control Bits – Output Data 36
Control Bits – Input Data 36
Programming Routine 36
Saving an Assembled Move in Flash 36
Indexed Moves 37
Controlling Moves In Progress 38
Find Home Moves 38
Jog Moves 38
Registration Moves 38
Absolute, Relative and Encoder Moves 38
Assembled Moves 38
Stall Detection with SMD23/4E-E Units 39
Idle Current Reduction 39

Chapter 3: Homing The SMD23/4E

Definition of Home Position 41
Position Preset 41
±Find Home Commands 41
Homing Inputs 41
Physical Inputs 41
Backplane Inputs 41
Homing Configurations 42
Homing Profiles 42
Home Input Only Profile 42
Profile with Network Home Proximity Bit 43
Profile with Overtravel Limit 44

Chapter 4: Installing the SMD23/4E

Location 45
Safe Handling Guidelines 45
Prevent Electrostatic Damage 45
Prevent Debris From Entering the Unit 45
Remove Power Before Servicing in a Hazardous Environment 45
Mounting 46
SMD23E Outline Drawing 46
SMD24E Outline Drawing 47
Connecting the Load 47
Network Connection 48
Network Status LED's 48
I/O Connector Pin Out 48
Power Wiring 48
Input Wiring 49

Chapter 5: Installing the SMD23/4E-M12

Location 51
Safe Handling Guidelines 51
Prevent Electrostatic Damage 51
Prevent Debris From Entering the Unit 51
Remove Power Before Servicing in a Hazardous Environment 51
Mounting 52
SMD23E-M12 Outline Drawing ... 52
SMD24E-M12 Outline Drawing ... 53
SMD23/4E-M12 Mounting 53
SMD23/4E-M12P Mounting 53
Connecting the Load 53
Network Connector 54
Compatible Connectors and Cordsets 54
Connectors 54
Ethernet Cordsets 54
TIA/EIA-568 Color Codes 55
Input Connector 55
Power Wiring 56
Input Wiring 57

Table of Contents

Chapter 6: Network Configuration

Factory Default 59
Assumptions 59
Firewalls 59
Installing the AMCI NET Software 59
Configure Your Network Interface Card (NIC) 60
Warning When Multiple NIC's Are On Your Computer 60
Attach the SMD23/4E 61
Using the AMCI Net Configurator 61
Start the Software 61
Connect to your SMD23/4E 62
The Meaning of the Product Name 63
Changing the IP Address 63
Setting Communications Protocol 63
Cycle Power to the SMD23/4E 63
Testing the New Address 63
SMD23E Tab 64
Using the Ping Command 65

Chapter 7: EtherNet/IP Configuration

RSLogix 5000 Configuration 67
Configure Bridge Module (As Needed) 67
Configuring a Built-in Ethernet Port (As Needed) 68
Adding the SMD23/4E 69
RSLogix 500 Configuration 70
Using Message Instructions in a MicroLogix 1100 PLC 70
Troubleshooting 74

Chapter 8: Modbus TCP Configuration

SMD23/4E Memory Layout 75
Supported Modbus Functions 76
Supported Modbus Exceptions 76
Sample Modbus Configuration 77
Initial Configuration 77
Configuring the Ethernet Scanner 78
Adding the SMD23/4E 78
Download Project to PLC 82
View SMD23/4E Data 82

Chapter 9: Configuration Mode Data Format

Modes of Operation 85
Configuration Mode 85
Command Mode 85
Multi-Word Format 85
Output Data Format 86
Control Word Format 86
Configuration Word Format 88
Notes on Other Configuration Words 89
Input Data Format 89
Control Word Format (Word 0) 89
Invalid Configurations 90

Chapter 10:Command Mode Data Format

Multi-Word Format 91
Command Bits Must Transition 91
Output Data Format 91
Command Bits MSW 92
Command Bits LSW 93
Command Blocks 95
Absolute Move 95
Relative Move 95
Hold Move 96
Resume Move 96
Immediate Stop 97
+Find Home (CW) 97
–Find Home (CCW) 98
+Jog Move (CW) 98
+Registration Move (CW) 99
–Jog Move (CCW) 99
–Registration Move (CCW) 100
Preset Position 100
Reset Errors 101
Run Assembled Move 101
Preset Encoder Position 102
Programming Blocks 103
First Block 103
Segment Block 103
Input Data Format 104
Status Bits MSW Format 104
Status Bits LSW Format 106
Motor Position Data 107
Encoder Position Data 107
Captured Encoder Position Data 107
Notes on Clearing a Driver Fault 107

Chapter A: IP Address Setup With BOOTP

Initial Configuration 109
Using the RA BOOTP Server 109
Attach the SMD23/4E 109
Start Your BOOTP Server 109
Enabling BOOTP Protocol 110
Setting the IP Address 111
Disabling the BOOTP Protocol 112
Testing the New IP Address 112

About this Manual

Read this chapter to learn how to navigate through this manual and familiarize yourself with the conventions used in it. The last section of this chapter highlights the manual's remaining chapters and their target audience.

Audience

This manual explains the set-up, installation, and operation of AMCI's SMD23E and SMD24E Integrated Stepper Indexer/Driver/Motors. It is written for the engineer responsible for incorporating these units into a design, as well as the engineer or technician responsible for their actual installation.

Applicable Units

This manual applies to all of the units in the SMD23/4E family.

Model Number	Description
SMD23E-130	Size 23 motor, 130 oz-in holding torque
SMD23E-240	Size 23 motor, 240 oz-in holding torque
SMD24E-350	Size 24 motor, 350 oz-in holding torque
SMD23E-130E	Same as SMD23E-130 with an integrated encoder
SMD23E-240E	Same as SMD23E-240 with an integrated encoder
SMD24E-350E	Same as SMD24E-350 with an integrated encoder
SMD23E-130-M12	Same as SMD23E-130 with sealed M12 connectors for IP50 rating.
SMD23E-240-M12	Same as SMD23E-240 with sealed M12 connectors for IP50 rating.
SMD24E-350-M12	Same as SMD24E-350 with sealed M12 connectors for IP50 rating.
SMD23E-130E-M12	Same as SMD23E-130E with sealed M12 connectors for IP50 rating.
SMD23E-240E-M12	Same as SMD23E-240E with sealed M12 connectors for IP50 rating.
SMD24E-350E-M12	Same as SMD24E-350E with sealed M12 connectors for IP50 rating.
SMD23E-130-M12P	Same as SMD23E-130-M12 with FDA approved clear coating, shaft and wire seals for IP67 rating.
SMD23E-240-M12P	Same as SMD23E-240-M12 with FDA approved clear coating, shaft and wire seals for IP67 rating.
SMD24E-350-M12P	Same as SMD24E-350-M12 with FDA approved clear coating, shaft and wire seals for IP67 rating.
SMD23E-130E-M12P	Same as SMD23E-130E-M12 with FDA approved clear coating, shaft and wire seals for IP67 rating.
SMD23E-240E-M12P	Same as SMD23E-240E-M12 with FDA approved clear coating, shaft and wire seals for IP67 rating.
SMD24E-350E-M12P	Same as SMD24E-350E-M12 with FDA approved clear coating, shaft and wire seals for IP67 rating.

Part Number Description

NOTE In general, the only difference between the SMD23 and SMD24 units are the outline drawings. They are functionally equivalent. When information in this manual applies to all of the units in the families and the name "SMD23/4E" is used. When a part of this manual only applies to units with encoders, the name "SMD23/4P-E" is used. When a part of this manual only applies to sealed units with M12 connectors, the name "SMD23/4E-M12" is used.

Revision Record

This manual, 940-0S134, is the fifth release of this manual. It was first released March 28th, 2014. It added information on the SMD24E units.

Revision History

940-0S133	Fourth Release.	September 10, 2013. Changed IP rating of dust tight unit from IP54 to IP50.
940-0S132	Third Release.	January 23, 2013. Add information on sealed M12 connectors
940-0S131	Second Release.	June 19, 2012. It corrects information on Ethernet LED's, optocoupler specifications, and adds torque curves.
940-0S130	Initial Release.	

Navigating this Manual

This manual is designed to be used in both printed and on-line formats. Its on-line form is a PDF document, which requires Adobe Acrobat Reader version 7.0+ to open it. The manual is laid out with an even number of pages in each chapter. This makes it easier to print a chapter to a duplex (double sided) printer.

Bookmarks of all the chapter names, section headings, and sub-headings were created in the PDF file to help navigate it. The bookmarks should have appeared when you opened the file. If they didn't, press the F5 key on Windows platforms to bring them up.

Throughout this manual you will also find ***blue text that functions as a hyperlink*** in HTML documents. Clicking on the text will immediately jump you to the referenced section of the manual. If you are reading a printed manual, most links include page numbers.

The PDF file is password protected to prevent changes to the document. You are allowed to select and copy sections for use in other documents and, if you own Adobe Acrobat version 7.0 or later, you are allowed to add notes and annotations.

Manual Conventions

Three icons are used to highlight important information in the manual:

NOTE **NOTES** highlight important concepts, decisions you must make, or the implications of those decisions.

CAUTION **CAUTIONS** tell you when equipment may be damaged if the procedure is not followed properly.

WARNING **WARNINGS** tell you when people may be hurt or equipment may be damaged if the procedure is not followed properly.

The following table shows the text formatting conventions:

Format	Description
Normal Font	Font used throughout this manual.
Emphasis Font	Font used the first time a new term is introduced.
Cross Reference	When viewing the PDF version of the manual, clicking on the cross reference text jumps you to referenced section.
HTML Reference	When viewing the PDF version of the manual, clicking on the HTML reference text will open your default web browser to the referenced web page.

Trademark Notices

The AMCI logo is a trademark, and "AMCI" is a registered trademark of Advanced Micro Controls Inc. All other trademarks contained herein are the property of their respective holders.

Where To Go From Here

This manual contains information that is of interest to everyone from engineers to operators. The table below gives a brief description of each chapter's contents to help you find the information you need to do your job.

CHP Num.	Chapter Title	Intended Audience
1	*Introduction to the SMD23/4E*	Anyone new to the SMD23/4E. This chapter gives a basic overview of the features available on the unit, typical applications, and electrical specifications.
2	*Move Profiles*	Anyone that needs detailed information on how the unit can be used to control motion in your application.
3	*Homing The SMD23/4E*	Anyone that needs detailed information on how the home position of the SMD23/4E can be set.
4	*Installing the SMD23/4E*	Anyone that must install an SMD23/4E on a machine. Includes information on mounting, grounding, and wiring specific to the units.
5	*Installing the SMD23/4E-M12*	Anyone that must install an SMD23/4E/M12 on a machine. Includes information on mounting, grounding, and wiring specific to the units.
6	*Network Configuration*	Anyone that must set the IP address and communications protocol of the SMD23/4E using the Ethernet Configurator utility from AMCI.
7	*EtherNet/IP Configuration*	Anyone that needs information on configuring an EtherNet/IP host. Detailed instructions are given for ControlLogix, CompactLogix, and MicroLogix systems from Rockwell Automation.
8	*Modbus TCP Configuration*	Anyone that needs information on configuring a Modbus TCP host.
9	*Configuration Mode Data Format*	Anyone that needs detailed information on the format of the network data to and from the SMD23/4E used to configure the unit.
10	*Command Mode Data Format*	Anyone that needs detailed information on the format of the network data to and from the SMD23/4E used to command the unit.
A	*IP Address Setup With BOOTP*	Information on setting IP address information using a BOOTP server.

Notes

CHAPTER 1

INTRODUCTION TO THE SMD23/4E

This manual is designed to get you quickly up and running with the SMD23/4E Integrated Stepper Indexer/Driver/Motor. As such, it assumes you have a basic understanding of stepper systems, such as the resolution you want to run your motor at, and the reasons why you'd want to use Idle Current Reduction and the reasons why you wouldn't. If these terms or ideas are new to you, we're here to help. AMCI has a great deal of information on our website and we are adding more all the time. If you can't find what you're looking for at http://www.amci.com, send us an e-mail or call us. We're here to support you with all of our knowledge and experience.

The SMD23/4E Family

The SMD23/4E is a growing product line from AMCI with a simple concept: a low-cost, stepper indexer, driver, and motor that can be attached to any popular industrial network. Each SMD23/4E attaches to an Ethernet network and communicates with either the EtherNet/IP or Modbus TCP protocols. The SMD23/4E units appear as a network node to your controller.

There are presently eighteen members of the SMD23/4E family:

- SMD23E-130: 130 oz-in motor
- SMD23E-240: 240 oz-in motor
- SMD24E-350: 350 oz-in motor
- SMD23E-130E: 130 oz-in motor with built in encoder
- SMD23E-240E: 240 oz-in motor with built in encoder
- SMD24E-350E: 350 oz-in motor with built in encoder

The built-in 1,024 count/turn encoder gives you additional functionality, such as position verification, moves based on encoder position, and stall detection.

Figure 1.1 SMD23E

If environmental protection is important, the six units above can be ordered with M12 connectors for power, I/O, and Ethernet connections. This units have an IP rating of IP50:

- SMD23E-130-M12
- SMD23E-240-M12
- SMD24E-350-M12
- SMD23E-130E-M12
- SMD23E-240E-M12
- SMD24E-350E-M12

Finally, if an IP67 rating is important, the M12 units can be ordered with a shaft seal, wire potting, and an FDA approved clear epoxy by appending "P" to the part number.

- SMD23E-130-M12P
- SMD23E-240-M12P
- SMD24E-350-M12P
- SMD23E-130E-M12P
- SMD23E-240E-M12P
- SMD24E-350E-M12P

The SMD23/4E Family (continued)

General Functionality

Each member of the SMD23/4E family has three integrated parts:

- An indexer that accepts commands over an Ethernet connection using the EtherNet/IP or Modbus TCP protocol
- A 3.4Arms micro-stepping driver that accepts 24 to 48Vdc as its input power source
- A high torque size 23 or size 24 stepper motor (130, 240, or 250 oz-in holding torque)

An incremental encoder is also available for applications that require additional position feedback.

The availability of the Ethernet/IP and Modbus TCP protocols makes the SMD23/4E units easy to integrate into a wide variety of controller systems. This combination of host and driver gives you several advantages:

- Sophisticated I/O processing can be performed in the host (PLC or other controller) before sending commands to the SMD23/4E unit
- All motion logic is programmed in the host, eliminating the need to learn a separate motion control language
- Eliminating the separate indexer and driver lowers Total System Cost

An SMD23/4E is powered by a nominal 24 to 48Vdc power source, and can accept surge voltages of up to 60Vdc without damage. The output motor current is fully programmable from 1.0Arms to 3.4Arms which makes the SMD23/4E suitable to a wide range of applications. In addition to the Motor Current setting, the Motor Steps per Turn, Idle Current Reduction, and Anti-Resonance Circuit features are also fully programmable. If you have used other stepper indexer products from AMCI you will find programming an SMD23/4E to be very similar to these products.

The SMD23/4E contains a true RMS motor current control driver. This means that you will always receive the motor's rated torque regardless of the *Motor Steps/Turn* setting. (Drivers that control the peak current to the motor experience a 30% decrease in motor torque when microstepping a motor.) The SMD23/4E automatically switches from RMS to peak current control when the motor is idle to prevent overheating the motor.

The standard SMD23/4E units have three DC inputs that are used by the indexer. SMD23/4E units with M12 connectors have two DC inputs. Configuration data from the host sets the function of these inputs. Each input can be individually configured as a:

- CW or CCW Limit Switch
- Home Limit Switch
- Capture Position Input (Will capture encoder position on SMD23/4E-E units.)
- Stop Manual or Registration Move Input
- Start Indexer Move
- Emergency Stop Input
- General Purpose Input

Encoder Functionality

All SMD23/4E units can be ordered with an internal 1,024 count/turn incremental encoder. Using an encoder gives you the ability to:

- Make absolute and relative moves based on the encoder count instead of the stepper pulse count
- Detect motor stall conditions

Indexer Functionality

The table below lists the functionality offered by the indexer built into the SMD23/4E.

Feature	Description
Ethernet/IP or Modbus TCP	Allows easy setup and communication with a wide range of host controllers such as the latest PLC's from Allen-Bradley.
Programmable Inputs	Each of the inputs can be programmed as a Home Limit, Over Travel Limit, Capture Input, Manual Jog Stop, Start Indexer Move, E-Stop, or a General Purpose Input.
Programmable Parameters	Starting Speed, Running Speed, Acceleration, Deceleration, and Accel/Decel Types are fully programmable.
Homing	Allows you to set the machine to a known position. An SMD23/4E homes to a discrete input and can use a bit in the Network Data as a home proximity input.
Relative Move	Allows you to drive the motor a specific number of steps in either direction from the current location.
Absolute Move	Allows you to drive the motor from one known location to another known location.
Jog Move	Allows you to jog the motor in either direction based on an input bit from your host controller.
Registration Move	Allows you to jog the motor in either direction based on an input bit from your host controller. When a controlled stop is issued, the move will output a programmable number of steps before coming to a stop.
Encoder Move	SMD23/4E-E units that are ordered with an internal encoder can perform moves based on the encoder position instead of the motor position. The encoder position sets the point that the move begins to decelerate.
Blend Move	Allows you to perform a sequence of relative moves without stopping between them.
Dwell Move	Allows you to perform a sequence of relative moves with a stop between each move that has a programmable length of time. Used to create highly accurate move profiles that avoid network latency issues.
Indexer Move	Allows you to program a move that is held in memory. The move is run when one of the programmable inputs makes a transition.
Hold Move	Allows you to suspend a move, and optionally restart it, without losing your position value.
Resume Move	Allows you to restart a previously held move operation.
Immediate Stop	Allows you to immediately stop all motion if an error condition is detected by your host controller.
Stall Detection	When an SMD23/4E is purchased with the encoder option, the encoder can be used to verify motion when a move command is issued.

Table 1.1 Indexer Functionality

Driver Functionality

This table summarizes the features of the stepper motor driver portion of an SMD23/4E.

Feature	Benefits
RMS Current Control	RMS current control give an SMD23/4E the ability to drive the motor at its fully rated power when microstepping. Peak current controllers typically experience a 30% drop in power when microstepping a motor.
Programmable Motor Current	RMS current supplied to the motor can be programmed from 1.0 to 3.4 amps in 0.1 amp increments. Reducing the motor current to the minimum needed for your application will significantly reduce the motors operating temperature
Programmable Idle Current Reduction	Extends motor life by reducing the motor current when motion is not occurring. This extends the life of the motor by reducing its operating temperature.
Programmable Motor Steps/Turn	Allows you to scale your motor count to a real world value. (counts per inch, counts per degree, etc.)
Anti-Resonance Circuitry	This circuitry gives each SMD23/4E the ability to modify motor current waveforms to compensate for mechanical resonance in your system. This will give you smooth performance over the entire speed range of the motor.
Over Temperature Detection	An SMD23/4E sets a warning bit in the network data when the temperature of the unit approaches its safe operating threshold.
Over Temperature Protection	Protects your SMD23/4E from damage by removing power from the motor if the internal temperature of the driver exceeds a safe operating threshold.

Table 1.2 Driver Functionality

Available Discrete Inputs

The SMD23/4E has two or three discrete DC inputs that accept 3.5 to 27Vdc signals. (5 to 24Vdc nominal) They can be wired as differential, sinking, or sourcing inputs. How your SMD23/4E uses these inputs is fully programmable. The active state of each input is also programmable. Programming their active states allow them to act as Normally Open(NO) or Normally Closed(NC) contacts.

Home Input

Many applications require that the machine be brought to a known position before normal operation can begin. This is commonly called “homing” the machine or bringing the machine to its “home” position. An SMD23/4E allows you to define this starting position in two ways. The first is with a Position Preset command. The second is with a sensor mounted on the machine. When you define one of the inputs as the Home Input, you can issue commands to the SMD23/4E that will cause the unit to seek this sensor. How the SMD23/4E actually finds the Home sensor is described in the ***Homing The SMD23/4E*** chapter starting on page 41.

CW Limit Switch or CCW Limit Switch

Each input can be defined as a CW or CCW Limit Switch. When used this way, the inputs are used to define the limits of mechanical travel. For example, if you are moving in a clockwise direction and the CW Limit Switch activates, all motion will immediately stop. At this point, you will only be able to jog in the counter-clockwise direction.

Available Discrete Inputs (continued)

Start Indexer Move Input

Indexer Moves are programmed through the Network Data like every other move. The only difference is that Indexer Moves are not run until a Start Indexer Move Input makes a inactive-to-active state transition. This allows an SMD23/4E to run critically timed moves that cannot be reliably started from the network due to data transfer lags.

If the quadrature encoder is available and one of the discrete DC inputs is programmed as a Start Indexer Move Input, then the quadrature encoder position data will be captured whenever the DC input makes a transition. An inactive-to-active state transition on the DC input will also trigger an Indexer Move if one is pending.

Emergency Stop Input

When an input is defined as an Emergency Stop, or E-Stop Input, motion will immediately stop when this input becomes active. The driver remains enabled and power is supplied to the motor. Any type of move, including a Jog or Registration Move, cannot begin while this input is active.

Stop Jog or Registration Move Input

When an input is configured as a Stop Jog or Registration Move Input, triggering this input during a Jog Move or Registration Move will bring the move to a controlled stop. The controlled stop is triggered on an inactive-to-active state change on the input. Only Jog Moves and Registration Moves can be stopped this way, all other moves ignore this input.

If the quadrature encoder is available, the quadrature encoder position data will be captured when the DC input makes an inactive-to-active transition if it is configured as a Stop Jog or Registration Move Input. The encoder position data is not captured if a Jog or Registration Move is not in progress. If you want to capture encoder position data on every transition of a DC input, configure it as a Start Indexer Move Input.

Capture Encoder Position Input

As described in the ***Start Indexer Move Input*** and ***Stop Jog or Registration Move Input*** sections above, an SMD23/4E can be configured to capture the encoder position value on a transition of a discrete DC input.

General Purpose Input

If your application does not require all of the inputs, you can configure the unused inputs as General Purpose Inputs. The inputs are not used by the SMD23/4E, but the input state is reported in the network data so their state can be monitored by your host controller.

Status LED's

Each SMD23/4E has two status LED's that show module and network status. As shown in figure 1.2, these LED's are located on the rear cover. SMD23/4E's without the M12 connectors have an additional status LED on the RJ45 connector that shows the status of the physical Ethernet link.

Module Status (MS) LED

The Module LED is a bi-color red/green LED that show the general status of the unit.

- **Steady Green:** Unit OK
- **Steady Red:** An Overtemperature Fault exists.
- **Blinking Green:** Successful write to flash memory. Power must be cycled to the unit before additional commands can be written to it.
- **Blinking Red:** Failed write to flash memory. You must cycle power to the unit to clear this fault.
- **Alternating Red/Green:** Communications failure. There is a communications error between the main processor and the ethernet co-processor within the unit. You must cycle power to the SMD23/4E to attempt to clear this fault.

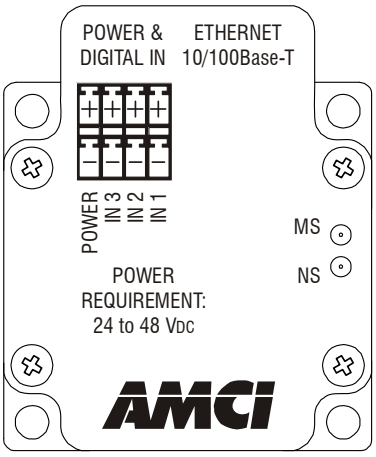


Figure 1.2 Rear Cover Status LED's

Network Status (NS) LED

The Network Status LED is a bi-color red/green LED. The state of the LED depends on the protocol the SMD23/4E is configured to for.

LED State	EtherNet/IP Definition	Modbus TCP Definition
Off	No Power	No power or no TCP connections
Alternating Red/Green	Power up Self-Test	Power up Self-Test
Flashing Green	Ethernet connection, but no CIP connections	Indicates number of concurrent connections with 2 second delay between group. The SMD23/4E supports up to 3 concurrent connections.
Steady Green	Valid Ethernet network and CIP connections	Should not occur. LED should always flash when network is connected.
Flashing Red	Network Connection Timeout	Not implemented in Modbus TCP
Steady Red	Duplicate IP address on network.	

Table 1.3 Network Status LED States

SMD23/4E Connectors

Input Connector

As shown in figure 1.3, the Input Connector is located on the top of the unit. Connections to the three differential digital inputs and power supply connections are made at this connector. The mating connector is supplied with the SMD23/4E. Spares are also available from AMCI under the part number MS-2X4. They are also available from Phoenix Contact under their part number 173 88 27.

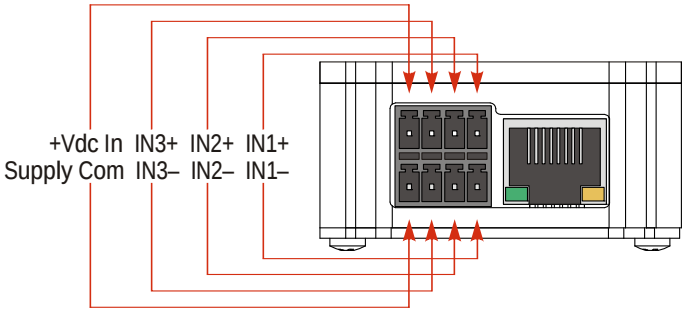


Figure 1.3 I/O Connector

Ethernet Port

The Ethernet Port is also located on the top of the SMD23/4E. The connector is a standard RJ-45 jack that will accept any standard 100baseT cable. The Ethernet port on the SMD23/4E is an "auto-sense" port that will automatically switch between 10baseT and 100baseT depending on the network equipment it is attached to. The port also has "auto switch" capability. This means that a standard cable can be used when connecting the SMD23/4E to any device, including a personal computer.

Link Status LED

This LED is on when there is a physical link between the Ethernet port of the SMD23/4E and the Ethernet port of the device the SMD23/4E is plugged into. This LED will flash when data is being transmitted over the Ethernet link.

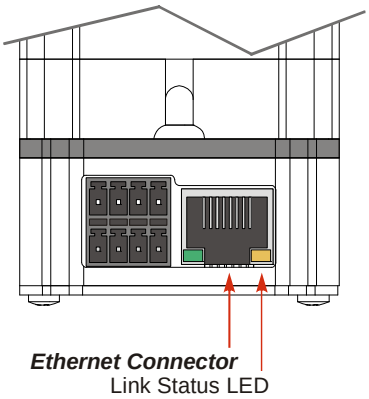


Figure 1.4 Ethernet Port Location

SMD23/4E-M12 Connectors

Input Connector

As shown in figure 1.5, the Input Connector is located on the back of the unit below the Ethernet Connector. All digital input and power supply connections are made at this connector. Figure 1.6 shows the pinout of the connector when viewed from the back of the SMD23/4E-M12.

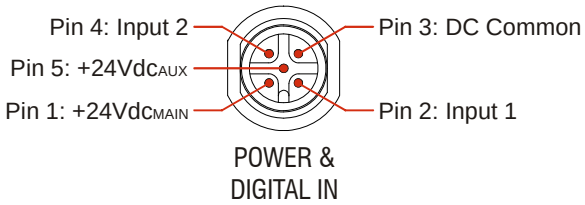


Figure 1.6 M12 Input Connector

Digital inputs on the SMD23/4E-M12 units are single ended and referenced to the DC Common pin. There are two power pins. $DCPower_{MAIN}$ powers both the control electronics and the motor. $DCPower_{AUX}$ powers only the control electronics. Using the $DCPower_{AUX}$ pin is optional. If your application requires you to cut power to your motor under some conditions, using the $DCPower_{AUX}$ pin allows you to cut power to your motor without losing your network connection. The connector is a standard five pin A-coded M12 connector that is rated to IP67 when the mate is properly attached.

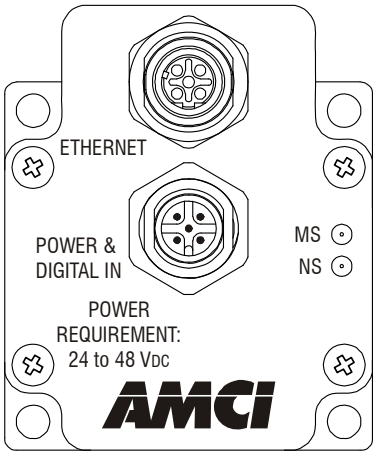


Figure 1.5 M12 Connector Locations

SMD23/4E-M12 Connectors (continued)

Ethernet Connector

Figure 1.5 also shows the placement of the sealed Ethernet Connector, while figure 1.7 shows the connector pinout when viewed from the back of the SMD23/4E-M12. The Ethernet port on the SMD23/4E is an "auto-sense" port that will automatically switch between 10baseT and 100baseT depending on the network equipment it is attached to. The port also has "auto switch" capability. This means that a standard cable can be used when connecting the SMD23/4E to any device, including a personal computer.

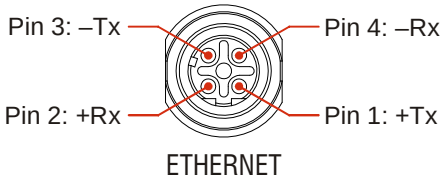


Figure 1.7 M12 Ethernet Connector Pinout

The connector is a standard four pin D-coded female M12 connector that is rated to IP67 when the mate is properly attached.

Compatible Connectors and Cordsets

Many different connectors and cordsets are available on the market, all of which will work with the SMD23/4E provided that the manufacturer follows the connector and Ethernet standards. AMCI has reviewed the following connectors and ethernet cordsets for compatibility with the SMD23/4E.

Connectors

AMCI #	Binder #	Description
MS-28	99-3729-810-04	Mating connector for Ethernet Connector. Male, 4 pin D-coded. Screw terminal connections. 6 to 8 mm dia. cable. Straight, IP67 rated when properly installed.
MS-31	99-0436-12-05	Mating connector for Power Connector. Female, 5 pin A-coded. Screw terminal connections. 6 to 8 mm dia. cable. Straight, IP67 rated when properly installed.

Table 1.4 Compatible Connectors

SMD23/4E-M12 Connectors (continued)

Compatible Connectors and Cordsets (continued)

Ethernet Cordsets

Phoenix Contact #	Description
1569391 SAC-4P-M12MSD/ 2,0-931	Bus system cable: 4-position, 26AWG, shielded. EIA/TIA 568B color coded Plug: Straight M12, D-coded, free conductor end. Cable length: 2 m
1569401 SAC-4P-M12MSD/ 5,0-931	Bus system cable: 4-position, 26AWG, shielded. EIA/TIA 568B color coded Plug: Straight M12, D-coded, free conductor end. Cable length: 5 m
1569414 SAC-4P-M12MSD/10,0-931	Bus system cable: 4-position, 26AWG, shielded. EIA/TIA 568B color coded Plug: Straight M12, D-coded, free conductor end. Cable length: 10 m
1569427 SAC-4P-M12MSD/15,0-931	Bus system cable: 4-position, 26AWG, shielded. EIA/TIA 568B color coded Plug: Straight M12, D-coded, free conductor end. Cable length: 15 m

Table 1.5 Ethernet Cordsets

NOTE These cordsets include the M12 connector, but the other end is un-terminated. This end can be punched down onto a patch panel, or an RJ45 connector can be added if the cordset is plugged directly into a switch.

Power Supply Sizing

An external 24 to 48Vdc power supply is required to power the control electronics and the stepper motor, so it must be rated to supply current to motor. The general rule of thumb is to specify a supply that can output the maximum current for your motor. The maximum motor current for the SMD23/4E is 3.4 Arms, but this can be limited by your software configuration.

By using the torque curves of your motor, it is possible to calculate the maximum power your motor will output and size your supply accordingly. If you use power calculations, double the calculated power when sizing your supply to account for power conversion losses and guarantee stall-free operation.

SMD23/4E-M12 units have an additional pin that supplies power to the electronics only. This allows you to remove power from the motor without losing the network connection.

Specifications

Driver Type

Two bipolar MOSFET H-bridges with 20KHz PWM current control.

Physical Dimensions

See page 46 for SMD23/4E or page 52 for SMD23/4E-M12

Weight

SMD23E-130(E) ..... 2.20 lbs. (1.00 kg.)
SMD23E-130(E)-M12(P) ..... 2.30 lbs. (1.05 kg.)
SMD23E-240(E) ..... 2.70 lbs. (1.23 kg.)
SMD23E-240(E)-M12(P) ..... 2.80 lbs. (1.27 kg.)
SMD24E-350(E) ..... 3.10 lbs. (1.41 kg.)
SMD24E-350(E)-M12(P) ..... 3.20 lbs. (1.45 kg.)

All weights are without mating connectors

Inputs

Electrical Characteristics: . . . . . .
SMD23/4E: Differential. 2500 Vac/dc opto-isolated. Can be wired as single ended inputs.

SMD23/4E-M12(P): Single ended sinking. 2500 Vac/dc opto-isolated.

DC Inputs accept 3.5 to 27Vdc without the need for an external current limiting resistor.

Faults

Faults are reported in the Network Input Data and can be cleared through the Network Output Data.

Motor Current

Programmable from 1.0 to 3.4Arms in 0.1 Amp steps.

$DCPower_{AUX}$ Current (SMD23/4E-M12 Units Only)

70 mA @ 24Vdc, 40mA @48Vdc

Motor Counts per Turn

Programmable to any value from 200 to 32,767 steps per revolution.

Incremental Encoder

Internal incremental encoder available as an option. Encoder supplies 1,024 counts per turn.

Idle Current Reduction

Programmable from 0% to 100% programmed motor current in 1% increments. Motor current is reduced to selected level if there is no motion for 1.5 seconds. Current is restored to full value when motion is started.

Environmental Specifications

Input Power ..... 24 to 48Vdc, surge to 60Vdc without damage to unit.

Ambient Operating Temperature

..... -4° to 122°F (-20° to 50°C)

Storage Temperature

..... -40° to 185°F (-40° to 85°C)

Humidity ..... 0 to 95%, non-condensing

IP Rating SMD23/4E: IP20
..... SMD23/4E-M12: IP50
..... SMD23/4E-M12P: IP67

Status LED's

See ***Status LED's*** section starting on page 16.

Connectors

All mating connectors are available separately under the following AMCI part numbers. The MS-2x4 is included with SMD23/4E units.

Connector	For Driver	AMCI Part #	Wire	Strip Length	Connection Type
I/O	SMD23/4E	MS-2x4	24 - 16 AWG	0.275 inches	Spring Cage Connector
I/O	SMD23/4E-M12	MS-31	18 AWG max.	0.197 inches	Screw Terminals
Ethernet	SMD23/4E-M12	MS-28	18 AWG max.	0.197 inches	Screw Terminals

Torque and Power Curves

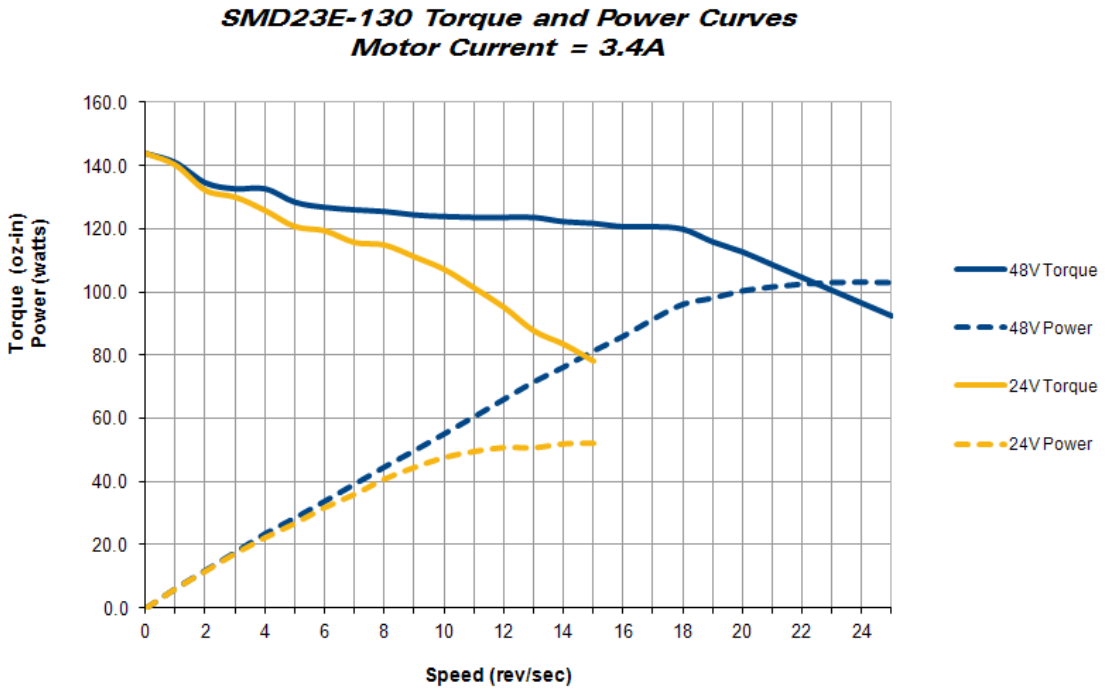


Figure 1.8 SMD23/4E-130 Torque and Power Curves

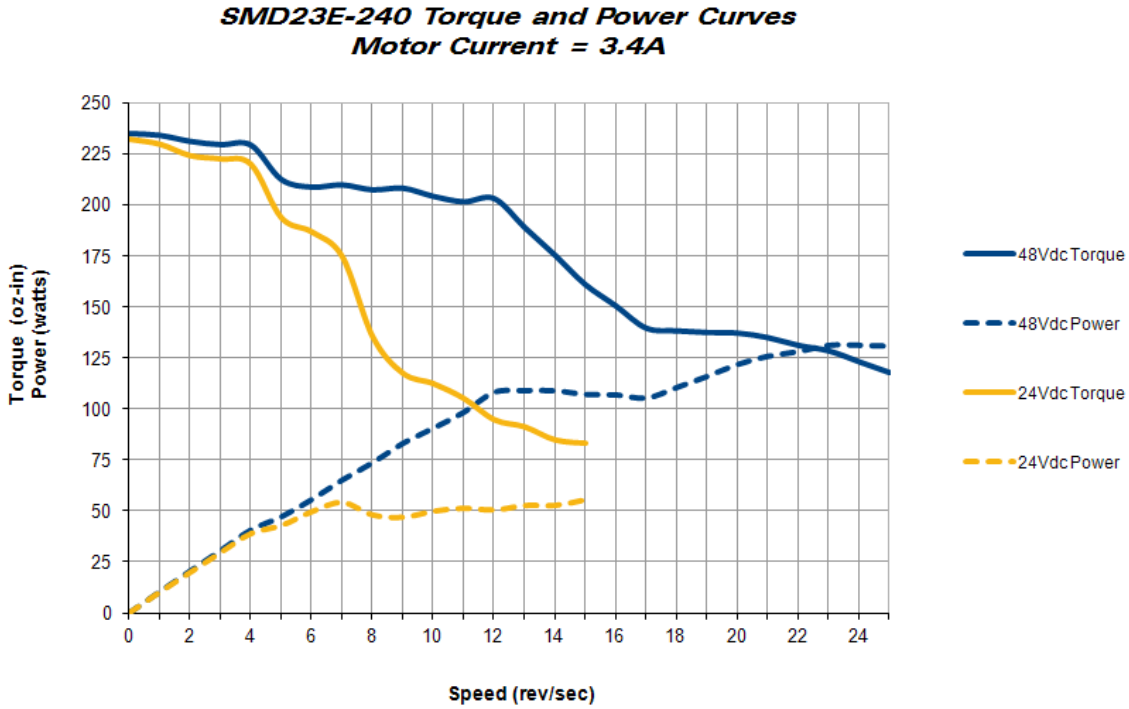


Figure 1.9 SMD23/4E-240 Torque and Power Curves

Torque and Power Curves (continued)

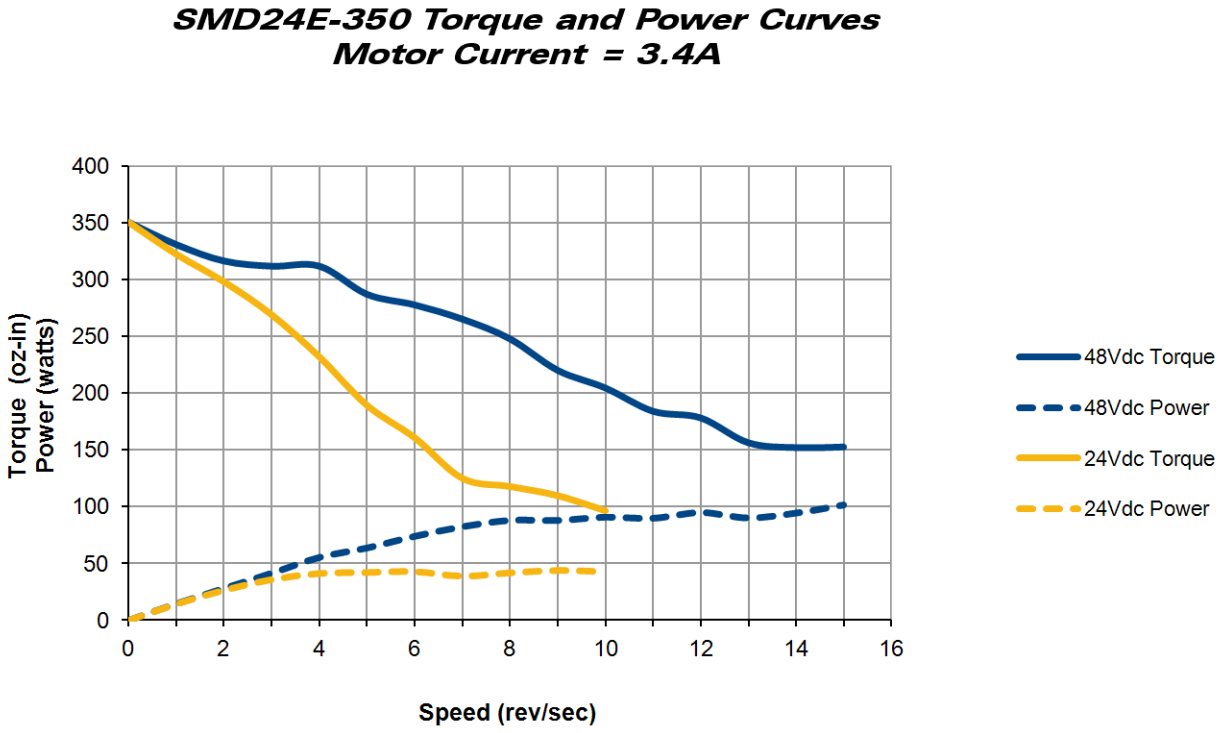


Figure 1.10 SMD24E-350 Torque and Power Curves

CHAPTER 2
MOVE PROFILES

When a move command is sent to an SMD23/4E, the unit calculates the entire profile before starting the move or issuing an error message. This chapter explains how the profiles are calculated and the different available moves.

Units of Measure

Distance: Every distance is measured in steps. Your programming will give you a specific number of steps needed to complete one rotation of the motor shaft. It is up to you to determine how many steps are required to travel the required distance in your application.

Speed: All speeds are measured in steps/second. Since the number of steps needed to complete one shaft rotation is determined by your programming, it is up to you to determine how many steps per second is required to rotate the motor shaft at your desired speed.

Acceleration: The typical unit of measure for acceleration and deceleration is steps/second/second, or steps/second^2. However, when programming an SMD23/4E, all acceleration and deceleration values must be programmed in the unit of measure of steps/millisecond/second.

- To convert from steps/second^2 to steps/millisecond/second, divide the value by 1000. This must be done when converting from a value used in the equations to a value programmed into an SMD23/4E.
- To convert from steps/millisecond/second to steps/second^2, multiply the value by 1000. This must be done when converting from the value programmed into an SMD23/4E to the value used in the equations.

Definition of Count Direction

Clockwise moves will always increase the motor position register that is reported back to the host. Some of the moves, such as the Jog Move, have a positive and negative command. A positive command, such as the +Jog Move command, will result in a clockwise rotation of the shaft.

Definition of Home Position

The Home Position is any position on your machine that you can sense and stop at. There are two ways to defining the Home Position. The first is using the Preset Position command to set the Motor Position register to a known value. The second method is using one of the ±Find Home commands. If you use the unit's ±*Find Home* commands, the motor position and encoder registers will automatically be set to zero once the home position is reached. Defining a Home Position is completely optional. Some applications, such as those that use the SMD23/4E for speed control, don't require position data at all.

Definition of Starting Speed

The Starting Speed is the speed that most moves will begin and end at. This value is set while configuring the unit and it has a valid range of 1 to 1,999,999 steps/second. This value is typically used to start the move above the motor's low frequency resonances and in micro-stepping applications to limit the amount of time needed for acceleration and deceleration. AMCI does not specify a default value in this manual because it is very dependent on motor size and attached load.

Definition of Target Position

The Target Position is position that you want the move to end at. There are two ways to define the Target Position, with relative coordinates or absolute coordinates.

Relative Coordinates

Relative coordinates define the Target Position as an offset from the present position of the motor. Most SMD23/4E moves use relative coordinates.

- The range of values for the Target Position when it is treated as an offset is ±8,388,607 counts. Positive offsets will result in clockwise moves, while negative offsets result in counter-clockwise moves.
- The Current Motor Position that is reported back to the host has a range of -32,768,000 to 32,767,999 counts. The only way to move beyond ±8,388,607 counts is with multiple relative moves or jog commands.

Absolute Coordinates

Absolute coordinates treat the Target Position as an actual position on the machine. Note that you must set the Home Position on the machine before you can run an Absolute Move. (See ***Definition of Home Position*** on the previous page.)

- The range of values for the Target Position when it is treated as an actual position on the machine is ±8,388,607 counts. The move will be clockwise if the Target Position is greater than the Current Position and negative if the Target Position is less than the Current Position.
- The Current Motor Position register that is reported back to the host has a range of -32,768,000 to 32,767,999 counts. However, you cannot move beyond ±8,388,607 counts with an Absolute Move. The only way to move beyond ±8,388,607 counts is with multiple relative moves or jog commands.

Definition of Acceleration Types

With the exception of Registration Moves, all move commands, including homing commands, allow you to define the acceleration type used during the move. The SMD23/4E supports three types of accelerations and decelerations. The type of acceleration used is controlled by the *Acceleration Jerk* parameter.

Linear Acceleration

When the Acceleration Jerk parameter equals zero, the axis accelerates (or decelerates) at a constant rate until the programmed speed is reached. This offers the fastest acceleration, but consideration must be given to insure the smoothest transition from rest to the acceleration phase of the move. The smoothest transition occurs when the configured Starting Speed is equal to the square root of the programmed Linear Acceleration. Note that other values will work correctly, but you may notice a quick change in velocity at the beginning of the acceleration phase.

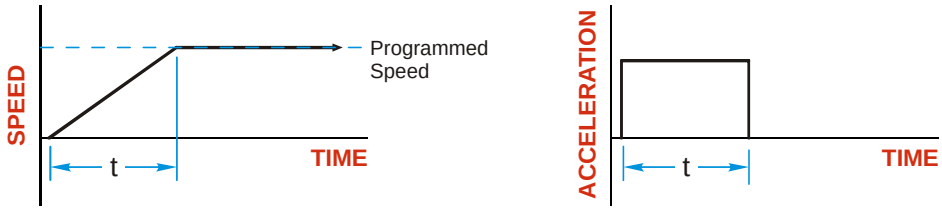


Figure 2.1 Linear Acceleration

Definition of Acceleration Types (continued)

Triangular S-Curve Acceleration

When the Acceleration Jerk parameter equals one, the axis accelerates (or decelerates) at a constantly changing rate that is slowest at the beginning and end of the acceleration phase of the move. The Triangular S-Curve type offers the smoothest acceleration, but it takes considerably longer than a Linear Acceleration to achieve the same velocity. Below is an example where the total time of acceleration is twice that of a linear acceleration move.

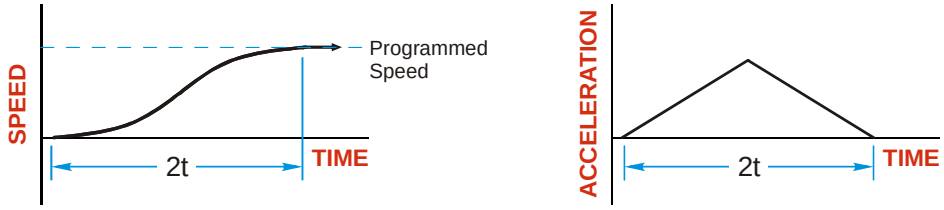


Figure 2.2 Triangular S-Curve Acceleration

Trapezoidal S-Curve Acceleration

When the Acceleration Jerk parameter is in the range of 2 to 5,000, Trapezoidal S-Curve acceleration is used. The Trapezoidal S-Curve acceleration is a good compromise between the speed of Linear acceleration and the smoothness of Triangular S-Curve acceleration. Like the Triangular S-Curve, this acceleration type begins and ends the acceleration phase smoothly, but the middle of the acceleration phase is linear. Figure 2.3 shows a trapezoidal curve when the linear acceleration phase is half of the total acceleration time. With this setting, the Trapezoidal S-Curve acceleration only requires 33% more time to achieve the same velocity as a Linear Acceleration.

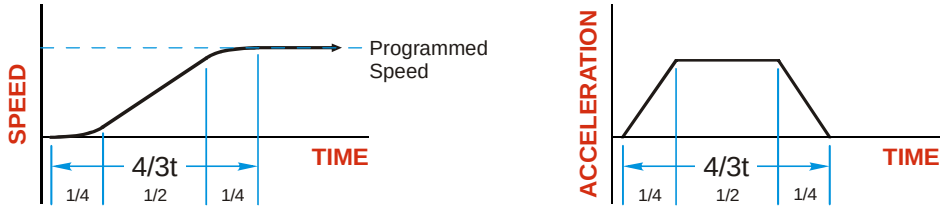


Figure 2.3 Trapezoidal S-Curve Acceleration

A Simple Move

As shown in the figure below, a move from A (Current Position) to B (Target Position) consists of several parts.

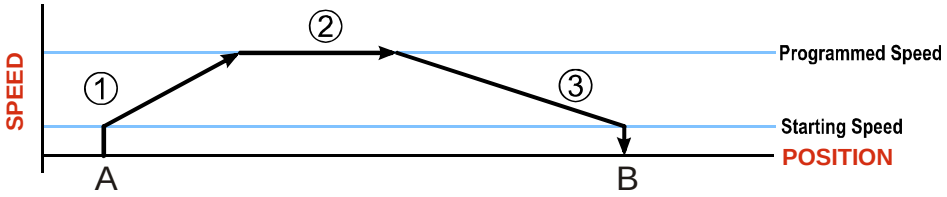


Figure 2.4 A Trapezoidal Profile

1) The move begins at point A, where the motor jumps from rest to the configured *Starting Speed*. The motor then accelerates at the programmed *Acceleration Value* until the speed of the motor reaches the *Programmed Speed*. Both the Acceleration Value and the Programmed Speed are programmed when the move command is sent to the SMD23/4E.
2) The motor continues to run at the Programmed Speed until it reaches the point where it must decelerate before reaching point B.
3) The motor decelerates at the *Deceleration Value*, which is also programmed by the move command, until the speed reaches the Starting Speed, which occurs at the Target Position (B). The motor stops at this point. Note that the acceleration and deceleration values can be different in the move.

Figure 2.4 above shows a Trapezoidal Profile. A Trapezoidal Profile occurs when the Programmed Speed is reached during the move. This occurs when the number of steps needed to accelerate and decelerate are less than the total number of steps in the move.

Figure 2.5 below shows a Triangular Profile. A Triangular Profile occurs when the number of steps needed to accelerate to the Programmed Speed and decelerate from the Programmed Speed are greater than the total number of steps in the move. In this case, the profile will accelerate as far as it can before it has to decelerate to reach the Starting Speed at the Target Position. The Programmed Speed is never reached.

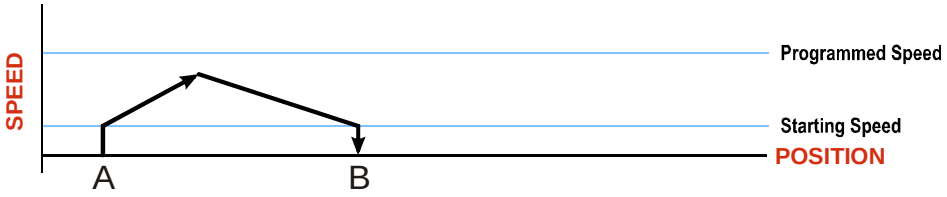


Figure 2.5 A Triangular Profile

Controlled and Immediate Stops

Once a move is started, there are several ways to stop the move before it comes to an end. These stops are broken down into two types:

- **Controlled Stop:** The axis immediately begins decelerating at the move's programmed deceleration value until it reaches the configured Starting Speed. The axis stops at this point. The motor position value is still considered valid after a Controlled Stop and the machine does not need to be homed again before Absolute Moves can be run.
- **Immediate Stop:** The axis immediately stops outputting pulses regardless of the speed the motor is running at. Because it is possible for the inertia of the load attached to the motor to pull the motor beyond the stopping point, the motor position value is considered invalid after an Immediate Stop and the machine must be homed again before Absolute Moves can be run.

Host Control

Hold Move Command: This command can be used with some moves to bring the axis to a Controlled Stop. The move can be resumed and finished, or it can be aborted. Not all moves are affected by this command. The section ***Basic Move Types***, starting on page 27, describes each move type in detail, including if the move is affected by this command.

Immediate Stop Command: When this command is issued from the host, the axis will come to an Immediate Stop. The move cannot be restarted and the machine must be homed again before Absolute Moves can be run.

Hardware Control

CW Limit and CCW Limit Inputs: In most cases, activating these inputs during a move will bring the axis to an Immediate Stop. The exceptions are the ±Find Home commands, the ±Jog Move commands, and the ±Registration Move commands. The ±Find Home commands are explained in chapter 3, ***Homing The SMD23/4E***, which starts on page 41. The ***±Jog Move*** commands are fully explained on page 29, and the ***±Registration Move*** commands are fully explained on page 31.

Emergency Stop Input: It is possible to configure an input as an Emergency Stop Input. When an Emergency Stop Input is activated, the axis will come to an Immediate Stop, regardless of the direction of travel. Note that power is not removed from the motor.

Basic Move Types

Relative Move

Relative Moves move an offset number of steps (n) from the Current Position (A). A trapezoidal profile is shown to the right, but Relative Moves can also generate triangular profiles. The command's Target Position is the move's offset. The offset can be in the range of ±8,388,607 counts. Positive offsets will result in clockwise moves, while negative offsets result in counter-clockwise moves.

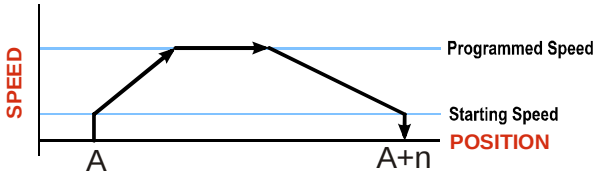


Figure 2.6 Relative Move

NOTE

1) You do not have to preset the position or home the machine before you can use a Relative Moves. That is, the Position Invalid status bit can be set.
2) Relative Moves allow you to move your machine without having to calculate absolute positions. If you are indexing a rotary table, you can preform a relative move of 30° multiple times without recalculating new positions in your controller. If you perform the same action with Absolute Moves, you would have to calculate your 30° position followed by your 60° position, followed by your 90° position, etc.

Basic Move Types (continued)

Relative Move (continued)

Relative Moves can be brought to a Controlled Stop by using the Hold Move Command from your host controller. When the command is accepted, the axis will immediately decelerate at the programmed rate and stop. When stopped successfully, the SMD23/4E will set a *Hold State* bit in the input data table. The Relative Move can be restarted with the Resume Move command from the host controller or the move can be aborted by starting another move. The Resume Move command allows you to change the move's Programmed Speed, Acceleration Value and Type, and the Deceleration Value and Type. The Target Position cannot be changed with the Resume Move Command.

Controlled Stop Conditions

- The move completes without error.
- You toggle the Hold Move control bit in the Network Output Data. Note that your holding position will most likely not be the final position you commanded. You can resume a held Relative Move by using the Resume Move command. The use of the Hold Move and Resume Move bits is further explained in the ***Controlling Moves In Progress*** section starting on page 38.

Immediate Stop Conditions

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Data.
- A positive transition on an input configured as an E-Stop Input.
- A CW/CWW Limit Switch is reached. If the limit that is reached is the same as the direction of travel, for example, hitting the CW limit while running a CW move, a *Reset Errors* command must be issued before moves are allowed in that direction again. If the limit that is reached is opposite the direction of travel, a *Reset Errors* command does not have to be issued.

Absolute Move

Absolute Moves move from the Current Position (A) to a given position (B). (The SMD23/4E calculates the direction and number of steps needed to move to the given position and moves that number of steps.) A trapezoidal profile is shown to the right, but Absolute Moves can also generate triangular profiles. The command's Target Position can be in the range of ±8,388,607 counts. The move will be clockwise if the Target Position is greater than the Current Position and counter-clockwise if the Target Position is less than the Current Position.

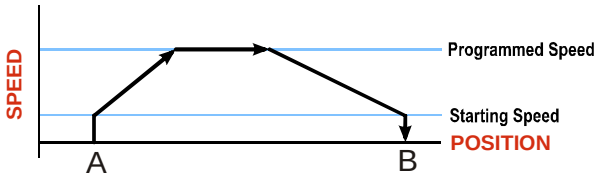


Figure 2.7 Absolute Move

NOTE

1) The *Home Position* of the machine must be set before running an Absolute Move. See chapter 3, ***Homing The SMD23/4E***, which starts on page 41, for information on homing the machine.
2) The Motor Position must be valid before you can use an Absolute Move. The Motor Position becomes valid when you preset the position or home the machine.
3) Absolute Moves allow you to move your machine without having to calculate relative positions. If you are controlling a rotary table, you can drive the table to any angle without having to calculate the distance to travel. For example an Absolute Move to 180° will move the table to the correct position regardless of where the move starts from.

Basic Move Types (continued)

Absolute Move (continued)

Controlled Stop Conditions

- The move completes without error.
- You toggle the Hold Move control bit in the Network Output Data. Note that your holding position will most likely not be the final position you commanded. You can resume a held Absolute Move by using the Resume Move bit or the move can be aborted by starting another move. The use of the Hold Move and Resume Move bits is explained in the ***Controlling Moves In Progress*** section starting on page 38.

Immediate Stop Conditions

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Data.
- A inactive-to-active transition on an input configured as an E-Stop Input.
- A CW/CWW Limit Switch is reached. If the limit that is reached is the same as the direction of travel, for example, hitting the CW limit while running a CW move, a *Reset Errors* command must be issued before moves are allowed in that direction again. If the limit that is reached is opposite the direction of travel, a *Reset Errors* command does not have to be issued.

±Jog Move

Jog Moves move in the programmed direction as long as the command is active. Two commands are available. The +Jog Move will rotate the shaft in a clockwise direction while the –Jog Move will rotate the shaft in a counter-clockwise direction. These commands are often used to give the operator manual control over the axis.

NOTE The Target Position value must be zero during a Jog Move. If it is not zero, you will perform a ***±Registration Move***, and the final stopping position may be unexpected.

Jog Moves are typically used to drive the machine under direct operator control, but they can also be used when you are interested in controlling the speed of the shaft instead of its position. One such application is driving a conveyor belt. In this application you are only interested in driving the conveyor at a specific speed, and you may need to vary the speed based on environmental conditions.

The CW Limit and CCW Limit inputs behave differently for ±Jog Moves and ±Registration Moves than all other move types. Like all moves, activating a limit will bring the move to an Immediate Stop. Unlike other moves, a Jog or Registration move can be started when an end limit switch is active provided that the commanded direction is opposite that of the activated switch. For example, a CW +Jog can be issued while the CCW limit switch is active. This allows you to move off of an activated end limit switch.

Basic Move Types (continued)

± Jog Move (continued)

As shown below, a Jog Moves begins at the programmed Starting Speed, accelerates at the programmed rate to the Programmed Speed and continues until a stop condition occurs. If it is a *Controlled Stop Condition*, the SMD23/4E will decelerate the motor to the starting speed and stop without losing position. If it is an *Immediate Stop Condition*, the motion stops immediately and the position becomes invalid.

It is possible to change the speed of a Jog Move without stopping the motion. The Programmed Speed, Acceleration, and Deceleration parameters can be changed during a Jog Move. When the Programmed Speed is changed, the motor will accelerate or decelerate to the new Programmed Speed using the new accelerate/decelerate parameter values. If you write a Programmed Speed to an SMD23/4E that is less than the starting speed, the Jog Move will continue at the previously programmed speed.

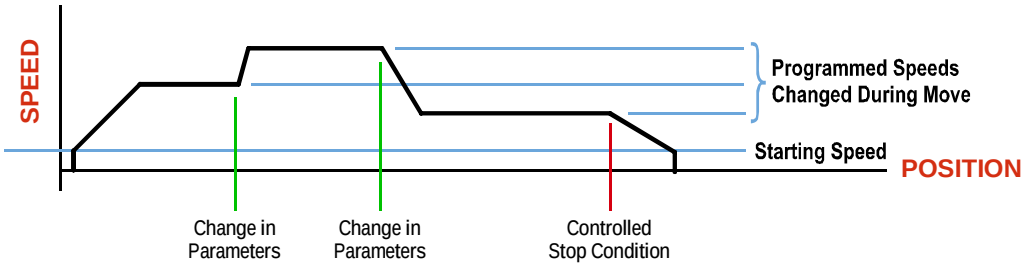


Figure 2.8 Jog Move

Controlled Stop Conditions

- The Jog Move Command bit is reset to "0".
- An inactive-to-active transition on an input configured as a *Stop Jog or Registration Move* Input.
- You toggle the Hold Move control bit in the Network Output Data. The use of the Hold Move and Resume Move bits is explained in the ***Controlling Moves In Progress*** section starting on page 38.

Immediate Stop Conditions

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Data.
- A inactive-to-active transition on an input configured as an E-Stop Input.
- A CW/CWW Limit Switch is reached. If the limit that is reached is the same as the direction of travel, for example, hitting the CW limit while running a CW move, a *Reset Errors* command must be issued before moves are allowed in that direction again. If the limit that is reached is opposite the direction of travel, a *Reset Errors* command does not have to be issued.

NOTE Note that it is possible to *start* a move while a CW or CCW Limit Switch is active as long as the direction of travel is *opposite* that of the activated Limit Switch. For example, it is possible to start a clockwise Jog Move while the CCW Limit Switch is active.

Basic Move Types (continued)

±Registration Move

Similar to a Jog Move, a Registration Move will travel in the programmed direction as long as the command is active. +Registration Moves result in clockwise rotation of the shaft, –Registration Moves result in counter-clockwise rotation of the shaft. When the command terminates under Controlled Stop conditions, the SMD23/4E will output a programmed number of steps as part of bringing the move to a stop. Note that all position values programmed with a Registration Move are relative values, not absolute machine positions.

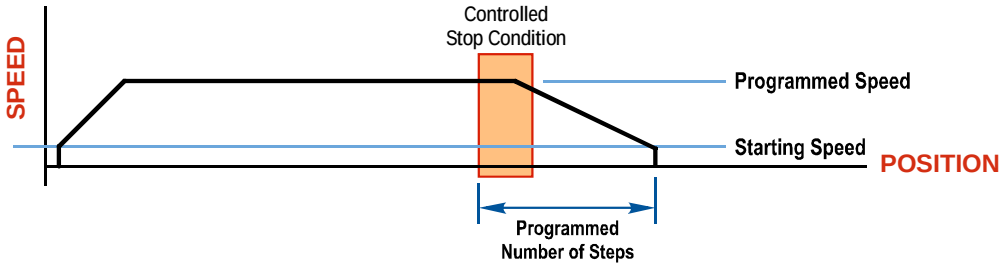


Figure 2.9 Registration Move

NOTE If the Programmed Number of Steps are less than the number of steps needed to bring the axis to a stop based on the Programmed Speed and Deceleration values set with the command, the SMD23/4E will decelerate at the programmed Deceleration value until it has output the Programmed Number of Steps and then stop the move without further deceleration.

An additional feature of the SMD23/4E is the ability to program the driver to ignore the Controlled Stop conditions until a minimum number of steps have occurred. This value is programmed through the Minimum Registration Move Distance parameter, which is set when you command the Registration Move. The figure below shows how the Minimum Registration Move Distance parameter effects when the Stop Condition is applied to the move. As shown in the second diagram, Controlled Stop conditions are level triggered, not edge triggered. If a Controlled Stop Condition occurs before the Minimum Registration Move Distance is reached and stays active, the move will begin its controlled stop once the Minimum Registration Move Distance is reached.

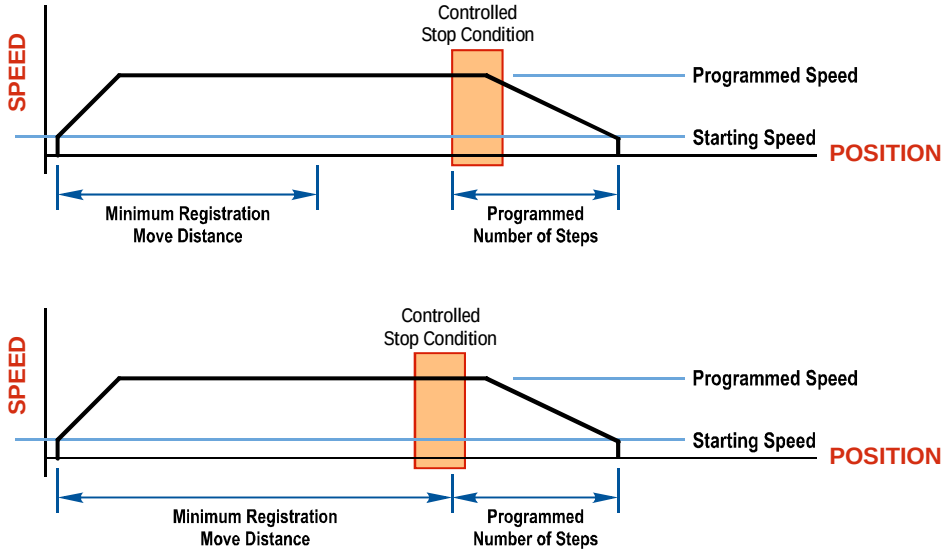


Figure 2.10 Min. Registration Move Distance

Basic Move Types (continued)

±Registration Move (continued)

Controlled Stop Conditions

- The Registration Move Command bit is reset to "0".
- A positive transition on an input configured as a *Stop Jog or Registration Move* Input.

NOTE Starting a Registration Move with a *Stop Jog or Registration Move* Input in its active state will result in a move of (*Minimum Registration Distance* + *Programmed Number of Steps*).

- You toggle the Hold Move control bit in the Network Output Data. The SMD23/4E responds by using the programmed Deceleration value to bring the move to a stop, without using the value of the Programmed Number of Steps parameter. A Registration Move does not go into the Hold State if the Hold Move control bit is used to stop the move and it cannot be restarted.

Immediate Stop Conditions

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Data.
- A positive transition on an input configured as an E-Stop Input.
- A CW/CWW Limit Switch is reached. If the limit that is reached is the same as the direction of travel, for example, hitting the CW limit while running a CW move, a *Reset Errors* command must be issued before moves are allowed in that direction again. If the limit that is reached is opposite the direction of travel, a *Reset Errors* command does not have to be issued.

NOTE Note that it is possible to *start* a move while a CW or CCW Limit Switch is active as long as the direction of travel is *opposite* that of the activated Limit Switch. For example, it is possible to start a clockwise Registration Move while the CCW Limit Switch is active.

Encoder Moves

When an SMD23/4E is ordered with an integral quadrature encoder, (SMD23/4E-xxxE part numbers), the position value from the encoder can be used to control the move instead of the position of the motor. Absolute and relative type moves are both supported.

NOTE You do not have the preset the position or home the machine before you can use a relative Encoder Move.

The figure below represents either a relative Encoder Move of 11,000 counts or an absolute Encoder Move to position 16,000. The figure shows that the encoder position you program in the move defines the point at which the motor begins to decelerate and stop. *It does not define the stopping position as it does in other move types.* The endpoint of the move depends on the speed of the motor when the programmed encoder position is reached and the deceleration values. This behavior is different from Absolute and Relative Moves where the position you program into the move is the end point of the move.

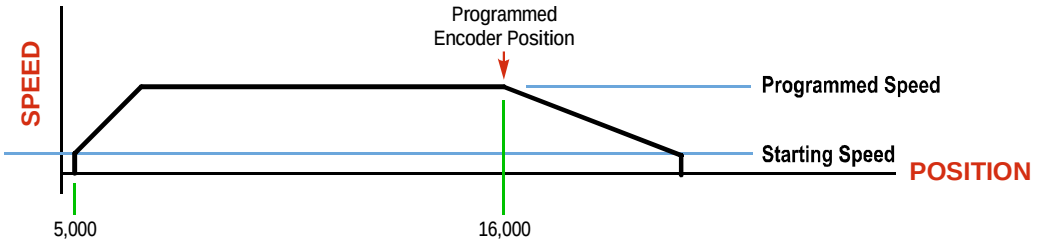


Figure 2.11 Encoder Move

Encoder Moves (continued)

Controlled Stop Conditions

- The move completes without error
- You toggle the Hold Move control bit in the Network Output Data. Note that your holding position will most likely not be the final position you commanded. You can resume a held Encoder Move by using the Resume Move bit or the move can be aborted by starting another move. The use of the Hold Move and Resume Move bits is explained in the ***Controlling Moves In Progress*** section starting on page 38.

Immediate Stop Conditions

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Data.
- A positive transition on an input configured as an E-Stop Input.
- A CW/CWW Limit Switch is reached. If the limit that is reached is the same as the direction of travel, for example, hitting the CW limit while running a CW move, a *Reset Errors* command must be issued before moves are allowed in that direction again. If the limit that is reached is opposite the direction of travel, a *Reset Errors* command does not have to be issued.

Assembled Moves

All of the moves explained so far must be run individually to their completion or must be stopped before another move can begin. The SMD23/4E also gives you the ability to pre-assemble more complex profiles from a series of relative moves that are then run with a single command. Each Assembled Move can consist of 2 to 16 segments. Two types of Assembled Moves exist in an SMD23/4E:

- **Blend Move -** A Blend Move gives you the ability to string multiple relative moves together and run all of them sequentially without stopping the shaft between moves. A Blend Move can be run in either direction, and the direction is set when the move command is issued.
- **Dwell Move -** A Dwell Move gives you the ability to string multiple relative moves together, and the SMD23/4E will stop between each move for a programed *Dwell Time*. Because motion stops between each segment, a Dwell Move allows you to reverse direction during the move.

Blend Moves

Each Relative Move defines a *segment* of the Blend Move. The following restrictions apply when programming Blend Moves.

1) Each segment of the Blend Move must be written to the SMD23/4E before the move can be initiated.
   - The SMD23/4E supports Blend Moves with up to sixteen segments.
2) Each segment is programmed as a relative move. Blend Moves cannot be programmed with absolute coordinates.
3) All segments run in the same direction. The sign of the target position is ignored and only the magnitude of the target position is used. The move's direction is controlled by the bit pattern used to start the move. If you want to reverse direction during your move, consider using the ***Dwell Move*** which is explained on page 35.
4) The Programmed Speed of each segment must be greater than or equal to the Starting Speed.
5) The Programmed Speed can be the same between segments. This allows you to chain two segments together.
6) For all segments except for the last one, the programmed position defines the end of the segment. For the last segment, the programmed position defines the end of the move.
7) Once you enter a segment, that segment's programmed acceleration and deceleration values are used to change the speed of the motor.
8) The blend segment must be long enough for the acceleration or deceleration portions of the segment to occur.

Assembled Moves (continued)

Blend Moves (continued)

The figure below shows a three segment Blend Move that is run twice. It is first run in the clockwise direction, and then in the counter-clockwise direction.

NOTE The deceleration value programmed with segment 3 is used twice in the segment. Once to decelerate from the Programmed Speed of segment 2 and once again to decelerate at the end of the move.

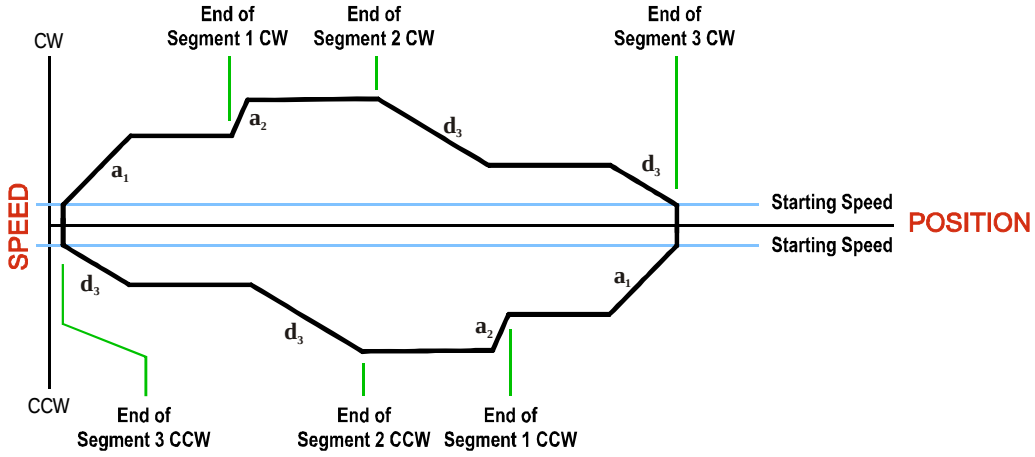


Figure 2.12 Blend Move

NOTE

1) You do not have to preset the position or home the machine before you can use a Blend Move. Because the Blend Move is based on Relative Moves, it can be run from any location.
2) The Blend Move is stored in the internal memory of the SMD23/4E and can be run multiple times once it is written to the unit. The Blend Move data stays in memory until power is removed, the unit is sent new Configuration Data, or a new Blend or Dwell Move is written to the unit. As described in ***Saving an Assembled Move in Flash*** on page 36, it is also possible to save a Blend Move to flash memory. This move is restored on power up and can be run as soon as you configure the SMD23/4E and enter Command Mode.
3) There are two control bits used to specify which direction the Blend Move is run in. This gives you the ability to run the Blend Move in either direction.

Controlled Stop Conditions

- The move completes without error.
- You toggle the Hold Move control bit in the Network Output Data. When this occurs, the SMD23/4E decelerates the move at the deceleration rate of the present segment to the Starting Speed and ends the move. Note that your final position will most likely not be the one you commanded. A Blend Move that is brought to a controlled stop with the Hold Move bit cannot be restarted. The use of the Hold Move bit is explained in the ***Controlling Moves In Progress*** section starting on page 38.

Immediate Stop Conditions

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Data.
- A positive transition on an input configured as an E-Stop Input.
- A CW/CWW Limit Switch is reached. If the limit that is reached is the same as the direction of travel, for example, hitting the CW limit while running a CW move, a *Reset Errors* command must be issued before moves are allowed in that direction again. If the limit that is reached is opposite the direction of travel, a *Reset Errors* command does not have to be issued.

Assembled Moves (continued)

Dwell Move

A Dwell Move gives you the ability to string multiple relative moves together and run all of them sequentially with a single start condition. Like a Blend Move, a Dwell Move is programmed into an SMD23/4E as a series of relative moves before the move is started.

Unlike a Blend Move, the motor is stopped between each segment of the Dwell Move for a programed *Dwell Time*. The Dwell Time is programmed as part of the command that starts the move. The Dwell Time is the same for all segments. Because the motor is stopped between segments, the motor direction can be reversed during the move. The sign of the target position for the segment determines the direction of motion for that segment. Positive segments will result in clockwise shaft rotation while a negative segment will result in a counter-clockwise shaft rotation. The following figure shows a drilling profile that enters the part in stages and reverses direction during the drilling operation so chips can be relieved from the bit.

You can accomplish this Dwell Move with a series of six relative moves that are sent down to the SMD23/4E sequentially. The two advantages of a Dwell Move in this case are that the SMD23/4E will be more accurate with the Dwell Time then you can be in your control program, and Dwell Moves simplify your program's logic.

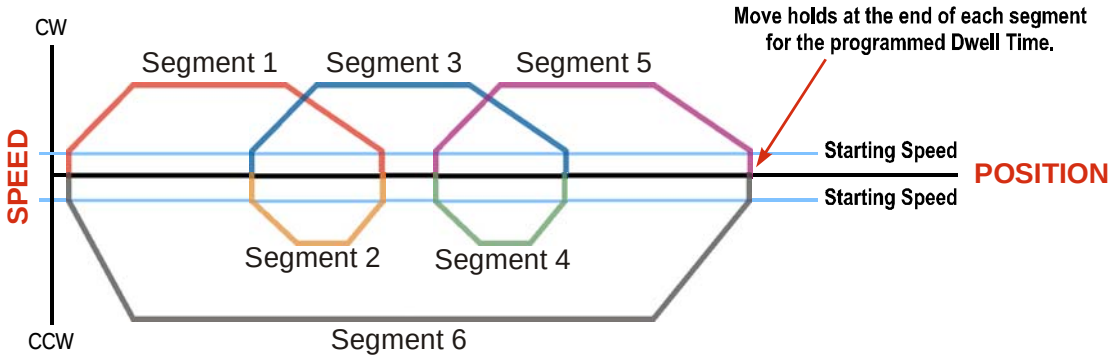


Figure 2.13 Dwell Move

NOTE

1) You do not have to preset the position or home the machine before you can use a Dwell Move. Because the Dwell Move is based on Relative Moves, it can be run from any location.

2) The Dwell Move is stored in the internal memory of an SMD23/4E and can be run multiple times once it is written to the unit. The Dwell Move data stays in memory until power is removed, the unit is sent new Configuration Data, or a new Blend or Dwell Move is written to the SMD23/4E. As described in ***Saving an Assembled Move in Flash*** on page 36, it is also possible to save a Dwell Move to flash memory. This move is restored on power up and can be run as soon as you configure your SMD23/4E and enter Command Mode.

Controlled Stop Conditions

- The move completes without error.
- You toggle the Hold Move control bit in the Network Output Data. When this occurs, the SMD23/4E decelerates the move at the deceleration rate of the present segment to the Starting Speed and ends the move. Note that your final position will most likely not be the one you commanded. A Dwell Move that is brought to a controlled stop with the Hold Move bit cannot be restarted.

Immediate Stop Conditions

- The Immediate Stop bit makes a 0 → 1 transition in the Network Input Data.
- A positive transition on an input configured as an E-Stop Input.
- A CW/CWW Limit Switch is reached. If the limit that is reached is the same as the direction of travel, for example, hitting the CW limit while running a CW move, a *Reset Errors* command must be issued before moves are allowed in that direction again. If the limit that is reached is opposite the direction of travel, a *Reset Errors* command does not have to be issued.

Assembled Move Programming

All of the segments in a Blend or Dwell Move must be written to the SMD23/4E before the move can be run. Segment programming is controlled with two bits in the Network Output Data and two bits in the Network Input Data. Blend and Dwell Moves are programmed in exactly the same way. When you start the move, a bit in the command data determines which type of Assembled Move is run. In the case of a Blend Move, the signs of the segment's Target Positions are ignored and all segments are run in the same direction. In the case of a Dwell Move, the signs of the segment's Target Positions determine the direction of the segment. For Dwell Moves, the Dwell Time is sent to the SMD23/4E as part of the command.

Control Bits – Output Data

- **Program Move bit –** Set this bit to tell the SMD23/4E that you want to program a Blend or Dwell Move Profile. The SMD23/4E will respond by setting the *Move Program Mode* bit in the Network Input Data. At the beginning of the programming cycle, the SMD23/4E will also set the *Transmit Move Segment* bit to signify that it is ready for the first segment.
- **Program Move Segment bit –** Set this bit to tell the SMD23/4E that the data for the next segment is available in the remaining data words.

Control Bits – Input Data

- **Move Program Mode bit –** The SMD23/4E sets this bit to tell you that it is ready to accept segment programming data in the remaining output data words. The actual transfer of segment data is controlled by the *Program Move Segment* and *Transmit Move Segment* bits.
- **Transmit Move Segment bit –** The SMD23/4E will set this bit to signal the host that it is ready to accept the data for the next segment.

Programming Routine

1) The host sets the *Program Move* bit in the Network Output Data.
2) The SMD23/4E responds by setting both the *Move Program Mode* and *Transmit Move Segment* bits in the Network Input Data.
3) When the host detects that the *Transmit Move Segment* bit is set, it writes the data for the first segment in the Network Output Data and sets the *Program Move Segment* bit.
4) The SMD23/4E checks the data, and when finished, resets the *Transmit Move Segment* bit. If an error is detected, it also sets the *Command Error* bit.
5) When the host detects that the *Transmit Move Segment* bit is reset, it resets the *Program Move Segment* bit.
6) The SMD23/4E detects that the *Program Move Segment* bit is reset, and sets the *Transmit Move Segment* bit to signal that it is ready to accept data for the next segment.
7) Steps 3 to 6 are repeated for the remaining segments until the entire move profile has been entered. The maximum number of segments per profile is 16.
8) After the last segment has been transferred, the host exits Assembled Move Programming Mode by resetting the *Program Move* bit.
9) The SMD23/4E resets the *Move Program Mode* bit and the *Transmit Blend Move Segment* bit.

Saving an Assembled Move in Flash

The SMD23/4E also contains the *Write Assembled Move* bit that allows you to store the Assembled Move in flash memory. This allows you to run the Assembled Move right after power up, without having to go through a programming sequence first. To use this bit, you follow the above programming routine with the *Write Assembled Move* bit set. When you reach step 9 in the sequence, the SMD23/4E responds by resetting the *Move Program Mode* and *Transmit Blend Move Segments* bits as usual and then it will flash the Status LED. If the LED is flashing green, the write to flash memory was successful. If it flashes red, then there was an error in writing the data. In either case, power must be cycled to the SMD23/4E before you can continue. With a limit of 10,000 write cycles, the design decision that requires you to cycle power to the SMD23/4E was made to prevent an application from damaging the module by continuously writing to it.

Indexed Moves

All of the moves that have been explained in the chapter up to this point can be started by a transition on one of the inputs instead of a command from the network. If the *Indexed Move* bit is set when the command is issued, the SMD23/4E will not run the move until the configured input makes an inactive-to-active transition. This allows you to run time critical moves that cannot be reliably started from the network because of messaging time delays.

- The input must be configured as a *Start Indexed Move Input.*
- The move begins with an inactive-to-active transition on the input. Note that an active-to-inactive transition on the input will not stop the move.
- The move command must stay in the Network Output Data while performing an Indexed Move. The move will not occur if you reset the command word before the input triggers the move.
- The move can be run multiple times as long as the move command data remains unchanged in the Network Output Data. The move will run on every inactive-to-active transition on the physical input if a move is not currently in progress. Once a move is triggered, the Start Indexed Move Input is ignored by the SMD23/4E until the triggered move is finished.
- As stated above, a move can be run multiple times as long at the move command data remains unchanged. If you wish to program a second move and run it as an Indexed Move type, then you must have a 0→1 transition on the move command bit before the new parameters are accepted. The easiest way to accomplish this is by writing a value of Ox0000 to the command word between issuing move commands.
- A Jog Move that is started as an Indexed Move will come to a controlled stop when the command bit in the Network Output Data is reset to zero.
- It is possible to perform an Indexed Registration Move by configuring two inputs for their respective functions. The first input, configured as a *Start Indexed Move Input*, starts the move and the second, configured as a *Stop Manual or Registration Move Input* causes the registration function to occur.
- You cannot issue a Hold Command with the Indexed Bit set and have the Hold Command trigger on the inactive-to-active transition of a physical input. Hold Commands are always acted upon as soon as they are accepted from the Network Output Data.
- You cannot issue an Immediate Stop Command with the Indexed Bit set and have the Immediate Stop Command trigger on the inactive-to-active transition of a physical input. Immediate Stop Commands are always acted upon as soon as they are accepted from the Network Output Data. If you need this functionality, consider programming the physical input as an E-Stop Input.
- You cannot issue a Clear Error Command with the Indexed Bit set and have the Clear Error Command trigger on the inactive-to-active transition of a physical input. Clear Error Commands are always acted upon as soon as they are accepted from the Network Output Data.

Controlling Moves In Progress

An SMD23/4E has the ability to place a running move on hold and later resume the move if an error did not occur while the move was in its Hold state. One potential application for this feature is bringing a move to a controlled stop when your controller senses an end-of-stock condition. The move can be put in its Hold state until the stock is replenished and then the move can be resumed.

Note that you do not have to resume a move once it has been placed in its Hold state. You can place a move in its Hold state to prematurely end the move with a controlled stop and issue a new move of any type from the stopped position.

The figure below shows a profile of a move that is placed in its Hold state and later resumed.

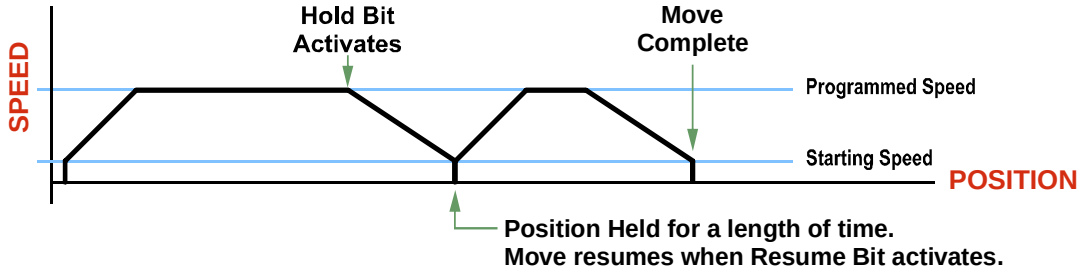


Figure 2.14 Hold/Resume a Move Profile

Find Home Moves

A Find Home command can be placed in a Hold state but cannot be resumed. This give you the ability to bring a Find Home command to a controlled stop if an error condition occurs.

Jog Moves

Jog Moves can be placed in a Hold state and resumed if error conditions, such as programming errors, have not occurred. New Acceleration, Deceleration, and Programmed Speed parameters can be written to the SMD23/4E while a Jog Move is in its hold state. If these parameters are accepted without error, the move can be resumed and it will use the new parameter values.

Registration Moves

Registration Moves can be brought to a controlled stop with the Hold bit, but they cannot be restarted.

Absolute, Relative and Encoder Moves

Absolute, Relative and Encoder Moves can be placed in a Hold state and resumed if error conditions, such as programming errors, have not occurred. New Acceleration, Deceleration, and Programmed Speed parameters can be written to the SMD23/4E while these moves are in their hold states. If the parameters are accepted without error, the move can be resumed and it will use the new parameter values. Note that a change to the Target Position is ignored.

Assembled Moves

A Blend or Dwell Move can be placed in a Hold state but cannot be resumed. This give you the ability to prematurely end an Assembled Move with a controlled stop. The Assembled Move is not erased from memory and can be run again without having to reprogram it.

Stall Detection with SMD23/4E-E Units

One of the additional features available to you when you order an SMD23/4E with an encoder is *Stall Detection*. When Stall Detection is enabled, the SMD23/4E-E monitors the encoder inputs for changes while a move is in progress. If the encoder inputs do not change as expected, the move stops and an error bit is reported to your host controller.

In order for the Stall Detection to work correctly, you must program the *Encoder Pulses/Turn* parameter to 1,024 in the Configuration Data of the SMD23/4E-E.

Idle Current Reduction

Idle Current Reduction allows you to prolong the life of your motor by reducing its idling temperature. Values for this parameter range from 0% (no holding torque when idle) to 100%.

Idle current reduction should be used whenever possible. By reducing the current, you are reducing the I^2R losses in the motor, which results in an exponential, not linear, drop in motor temperature. This means that even a small reduction in the idle current can have a significant effect on the temperature of the motor.

NOTE Note that the reduction values are "to" values, not "by" values. Setting a motor current to 2Arms and the current reduction to 25% will result in an idle current of 0.5Apk. (The SMD23/4E always switches from RMS to peak current control when the motor is idle to prevent motor damage due to excessive heating.)

Notes

CHAPTER 3
HOMING THE SMD23/4E

This chapter explains the various ways of homing the SMD23/4E. Inputs used to home the unit are introduced and diagrams that show how the unit responds to a homing command are given.

Definition of Home Position

The Home Position is any position on your machine that you can sense and stop at. Once at the Home Position, the motor position register of an SMD23/4E must be set to an appropriate value. If you use the unit's ±*Find Home* commands, the motor position register will automatically be set to zero once the home position is reached. The Encoder Position register will also be reset to zero if the quadrature encoder is available and enabled.

NOTE Defining a Home Position is completely optional. Some applications, such as those that use the SMD23/4E for speed control, don't require position data at all.

With the exception of Absolute Moves, an SMD23/4E can still perform all of its move commands if the Home Position is not defined.

Position Preset

One of the ways to define the Home Position is to issue the Preset Position command to the SMD23/4E. Before doing this, your host controller will need a way of sensing position. The machine position data must be brought into the host, the correct preset value calculated, and this value written to the SMD23/4E with the Position Preset command. The motor and encoder position values can be preset anywhere in the range of –8,388,607 to +8,388,607.

±Find Home Commands

The other choice is to use the module's Find Home commands to order the SMD23/4E to find the Home Position based on sensors brought into the unit. The +Find Home command begins searching by rotating the motor shaft clockwise and ends when the home sensor triggers while the SMD23/4E is rotating clockwise at the starting speed. The –Find Home command operates in the same way but starts and ends with counter-clockwise rotation.

Homing Inputs

Four inputs can be used when homing the module. These inputs are either physical inputs attached to the module or bits in the network output data words.

Physical Inputs

- **Home Input:** This input is used to define the actual home position of the machine.
- **CW Limit Switch Input:** This input is used to prevent overtravel in the clockwise direction.
- **CCW Limit Switch Input:** This input is used to prevent overtravel in the counter-clockwise direction.

Backplane Inputs

- **Home Proximity Bit:** The SMD23/4E can be configured to ignore changes on the physical homing input until the Home Proximity Bit makes a 0→1 transition. The SMD23/4E will home on the next inactive-to-active change on the physical input once this transition occurs. You must program your host to control the state of this bit.

Homing Configurations

An SMD23/4E must have one of its DC inputs configured as the home input before one of the ± *Find Home* commands can be issued.

NOTE

1) You do not have to configure and use CW or CCW Limits. If you choose to configure the module this way, then the SMD23/4E has no way to automatically prevent overtravel during a homing operation. You must prevent overtravel by some external means, or ensure that the homing command is issued in the direction that will result in reaching the homing input directly.
2) You can use a bit in the Network Output Data as a home proximity input. Using this bit is completely optional.

Homing Profiles

NOTE The +Find Home command is used in all of these examples. The –Find Home command will generate the same profiles in the opposite direction.

Home Input Only Profile

Figure 3.1 below shows the move profile generated by a +Find Home command when you use the Home Input without the Backplane Home Proximity bit.

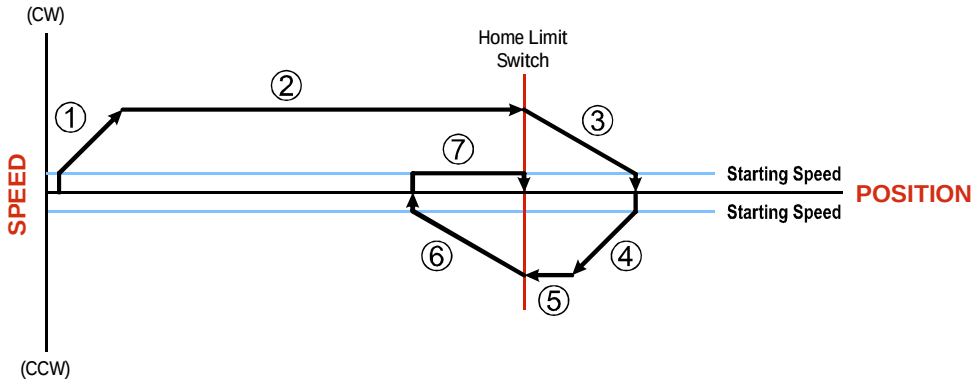


Figure 3.1 Home Input Profile

1) Acceleration from the configured Starting Speed to the Programmed Speed
2) Run at the Programmed Speed until the Home Input activates
3) Deceleration to the Starting Speed and stop, followed by a two second delay.
4) Acceleration to the Programmed Speed opposite to the requested direction.
5) Run opposite the requested direction until the Home Input transitions from Active to Inactive
6) Deceleration to the Starting Speed and stop, followed by a two second delay.
7) Return to the Home Input at the configured Starting Speed. Stop when the Home Input transitions from inactive to active.

NOTE If the Home Input is active when the command is issued, the move profile begins at step 5 above.

Homing Profiles (continued)

Profile with Network Home Proximity Bit

Figure 3.2 below shows the move profile generated by a +Find Home command when you use the Home Input with Network Home Proximity bit.

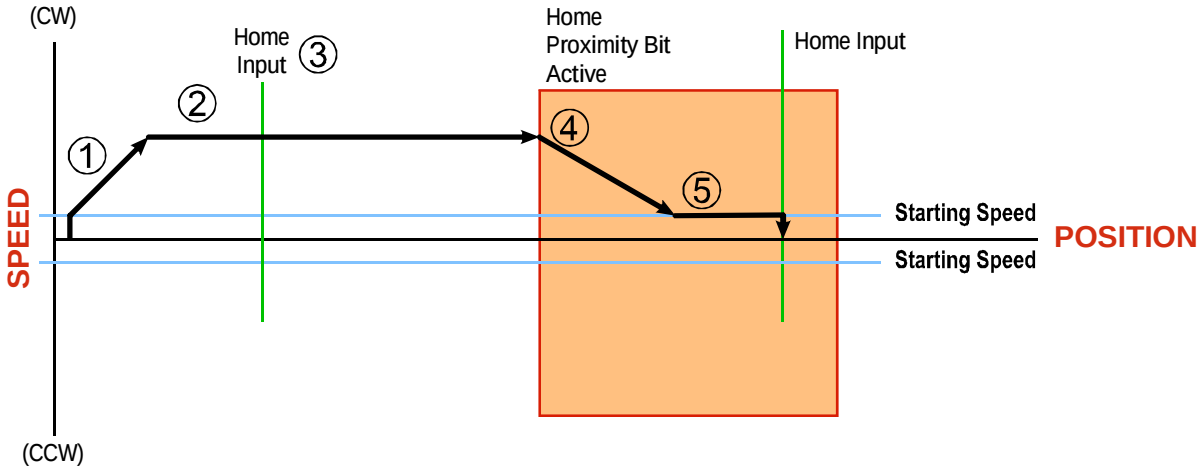


Figure 3.2 Homing with Proximity

1) Acceleration from the configured Starting Speed to the Programmed Speed
2) Run at the Programmed Speed
3) Ignores the Home Input because Home Proximity Bit has not made a 0→1 transition.
4) Deceleration towards the Starting Speed when the Home Proximity bit transitions from 0 to 1. The axis will stop as soon as the Home Input becomes active.
5) The Starting Speed is the minimum speed the profile will run at. If the axis decelerates to the Starting Speed before reaching the Home Input, it will continue at this speed.

NOTE Figure 3.2 shows the Backplane Home Proximity Bit staying active until the SMD23/4E reaches its home position. This is valid, but does not have to occur. As stated in step 4, the SMD23/4E starts to hunt for the home position as soon and the Backplane Home Proximity Bit makes a 0→1 transition

Homing Profiles (continued)

Profile with Overtravel Limit

Figure 3.3 below shows the move profile generated by a +Find Home command when you use:

- CW Overtravel Limit
- Home Input without Backplane Home Proximity Bit

The profile is generated when you encounter an overtravel limit in the direction of travel. (In this example, hitting the CW limit while traveling in the CW direction.) Hitting the overtravel limit associated with travel in the opposite direction is an Immediate Stop condition. The motor will stop all motion and issue a *Home Invalid* error to your host.

The SMD23/4E will stop with an error if both overtravel limits are activated while the unit is trying to find the home position.

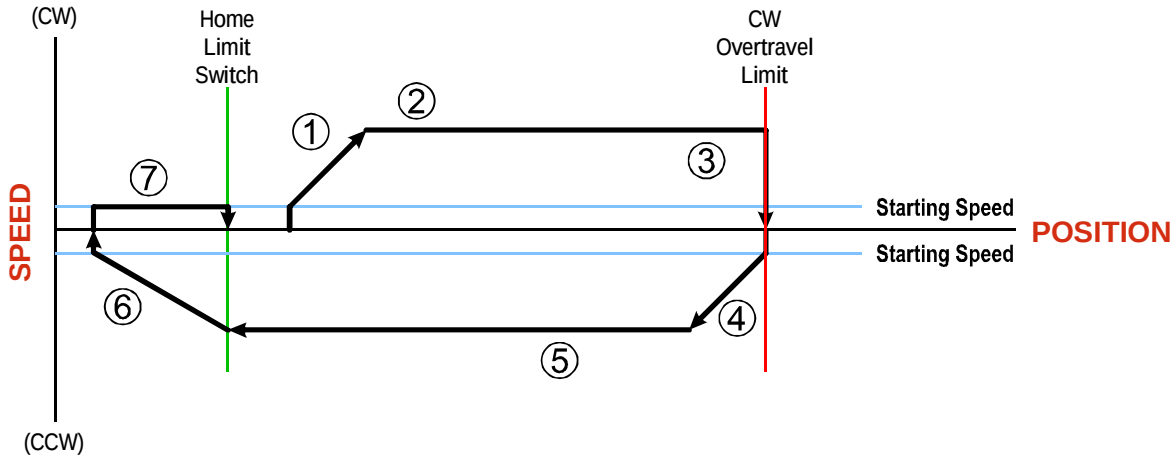


Figure 3.3 Profile with Overtravel Limit

1) Acceleration from the configured Starting Speed to the Programmed Speed
2) Run at the Programmed Speed
3) Hit CW Limit and immediately stop, followed by a two second delay.
4) Acceleration to the Programmed Speed opposite to the requested direction.
5) Run opposite the requested direction until the Home Input transitions from Active to Inactive
6) Deceleration to the Starting Speed and stop, followed by a two second delay.
7) Return to the Home Input at the configured Starting Speed. Stop when the Home Input transitions from Inactive to Active.

NOTE If the overtravel limit is active when the Find Home Command is active, the profile will begin at step 4.

CHAPTER 4
INSTALLING THE SMD23/4E

This chapter applies to the standard SMD23/4E units. If you are installing a sealed SMD23/4E that has M12 connectors, please refer to chapter 5 starting on page 51.

Location

SMD23/4E units are suitable for use in an industrial environment that meet the following criteria:

- Only non-conductive pollutants normally exist in the environment, but an occasional temporary conductivity caused by condensation is expected.
- Transient voltages are controlled and do not exceed the impulse voltage capability of the product's insulation.

Note that these criteria apply to the system as a whole, not just to the SMD23/4E.

These criteria are equivalent to the *Pollution Degree 2* and *Over Voltage Category II* designations of the International Electrotechnical Commission (IEC).

Safe Handling Guidelines

Prevent Electrostatic Damage

CAUTION Electrostatic discharge can damage the SMD23/4E. Follow these guidelines when handling the unit.

1) Touch a grounded object to discharge static potential before handling the unit.
2) Work in a static-safe environment whenever possible.
3) Wear an approved wrist-strap grounding device.
4) Do not touch the pins of the network connector or I/O connector.
5) Do not disassemble the unit
6) Store the unit in its anti-static bag and shipping box when it is not in use.

Prevent Debris From Entering the Unit

WARNING While mounting of all devices, be sure that all debris (metal chips, wire strands, tapping liquids, etc.) is prevented from falling into the unit. Debris may cause damage to the unit or unintended machine operation with possible personal injury.

Remove Power Before Servicing in a Hazardous Environment

WARNING Remove power before removing or installing any SMD23/4E units in a hazardous environment.

Mounting

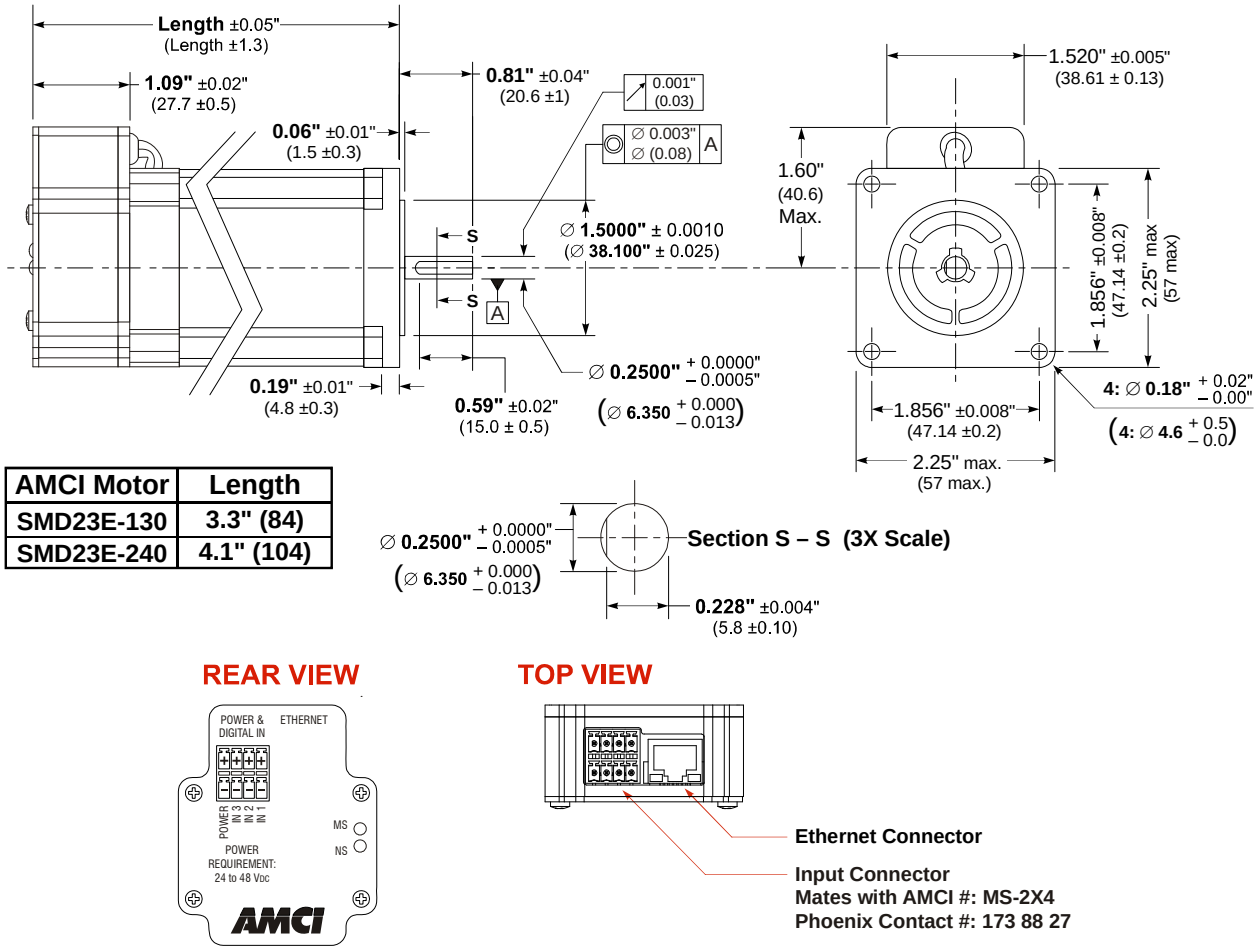
All AMCI motor have flanges on the front of the motor for mounting. This flange also acts as a heatsink, so motors should be mounted on a large, unpainted metal surface. Mounting a motor in this fashion will allow a significant amount of heat to be dissipated away from the motor, which will increase the unit's life by reducing its operating temperature. If you cannot mount the motor on a large metal surface, you may need to install a fan to force cooling air over the SMD23/4E.

Motors should be mounted using the heaviest hardware possible. AMCI motors can produce high torques and accelerations that may weaken and shear inadequate mounting hardware.

NOTE

1) The motor case must be grounded for proper operation. This is usually accomplished through its mounting hardware. If you suspect a problem with your installation, such as mounting the motor to a painted surface, then run a bonding wire from the motor to a solid earth ground point near it. Use a minimum #8 gauge stranded wire or 1/2" wire braid as the grounding wire
2) Do not disassemble *any* stepper motor. A significant reduction in motor performance will result.

SMD23E Outline Drawing



AMCI Motor	Length
SMD23E-130	3.3" (84)
SMD23E-240	4.1" (104)

Figure 4.1 SMD23E Outline Drawing

Mounting (continued)

SMD24E Outline Drawing

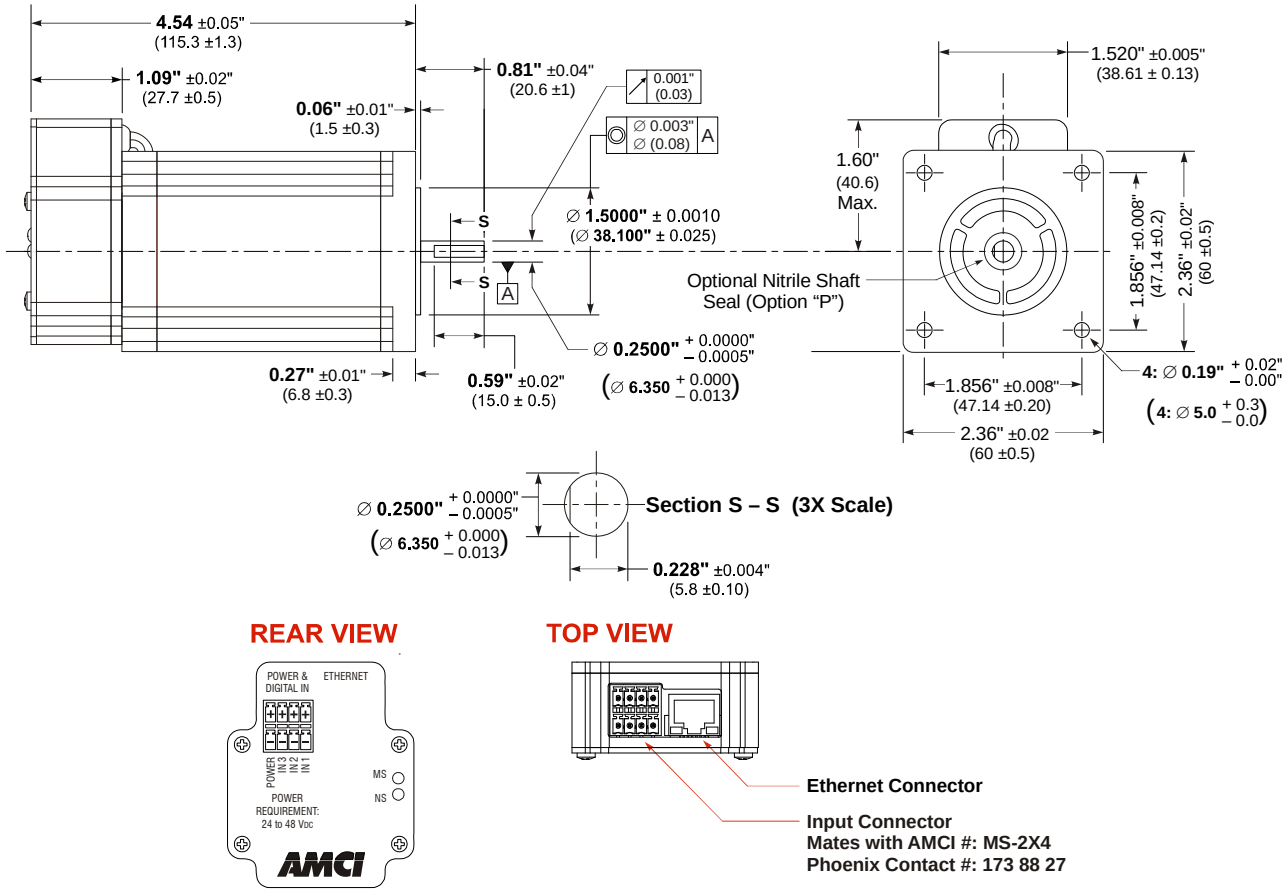


Figure 4.2 SMD24E Outline Drawing

Connecting the Load

Care must be exercised when connecting your load to the stepper motor. Even small shaft misalignments can cause large loading effects on the bearings of the motor and load. The use of a flexible coupler is *strongly* recommended whenever possible.

Network Connection

The Ethernet connector is located on the top of the SMD23/4E. The connector is a standard RJ-45 jack that will accept any standard 100baseT cable. Because the port can run at 100 Mbit speeds, Category 5, 5e, or 6 cable should be used.

The Ethernet port on the SMD23/4E has "auto switch" capability. This means that a standard cable can be used when connecting the SMD23/4E to any device. A crossover cable is not necessary when connecting an SMD23/4E directly to a PC.

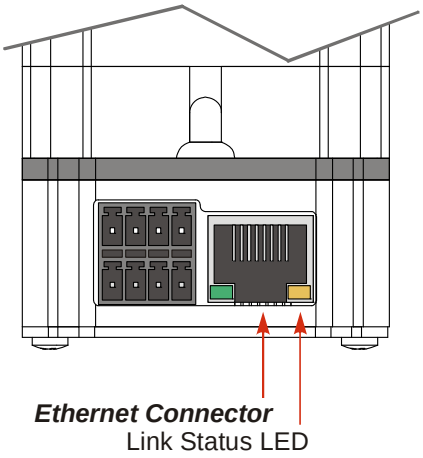


Figure 4.3 Ethernet Port Location

Network Status LED's

There are two LED's that indicate the health of the network connection. These are the Network LED located on the rear of the SMD23/4E and an LED that is built into the network connector. (See figure 4.1 on the previous page for the location of the Network LED.) The Status LED, which is also on the rear cover, indicates the state of the SMD23/4E itself.

- **Network Status (EtherNet/IP) –** This LED on the rear cover indicates the status of the connection between the SMD23/4E and its host.
- **Network Status (Modbus TCP) –** This LED on the rear cover indicates the number of TCP connections to the SMD23/4E.
- **Link Status –** On when there is a physical link between the Ethernet port of the SMD23/4E and the Ethernet port of the device the SMD23/4E is plugged into.

The Network Status LED's are fully described in the ***Status LED's*** section of this manual starting on page 16.

I/O Connector Pin Out

The I/O Connector is located on the top of the SMD23/4E. The mate for this connector is included with the unit. It is also available from AMCI under the part number MS-2X4 and Phoenix Contact under their part number 173 88 27. Figure 4.4 shows the pin out for the I/O connector.

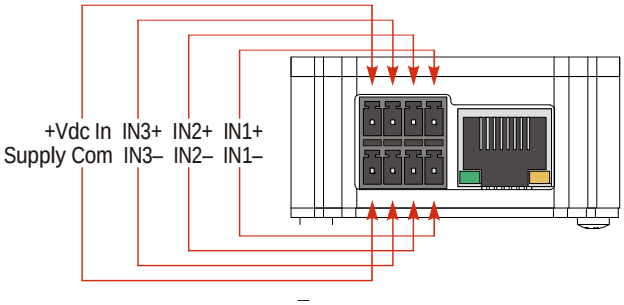


Figure 4.4 I/O Connector

Power Wiring

The SMD23/4E accepts 24 to 48Vdc as its input power. The power pins are the outer two pins of the MS-2X4 connector. See figure 4.4 for proper connections to the MS-2X4. The connector will accept 16 to 24 AWG wire. AMCI strongly suggests using 16 AWG wire to minimize power losses.

CAUTION Do not apply 120 Vac to any pins of the SMD23/4E. If this occurs, the unit will be damaged and you will void the unit's warranty.

Input Wiring

Figure 4.5 below shows how to wire discrete DC differential, sourcing, and sinking sensors to Inputs 1, 2, and 3 of the SMD23/4E.

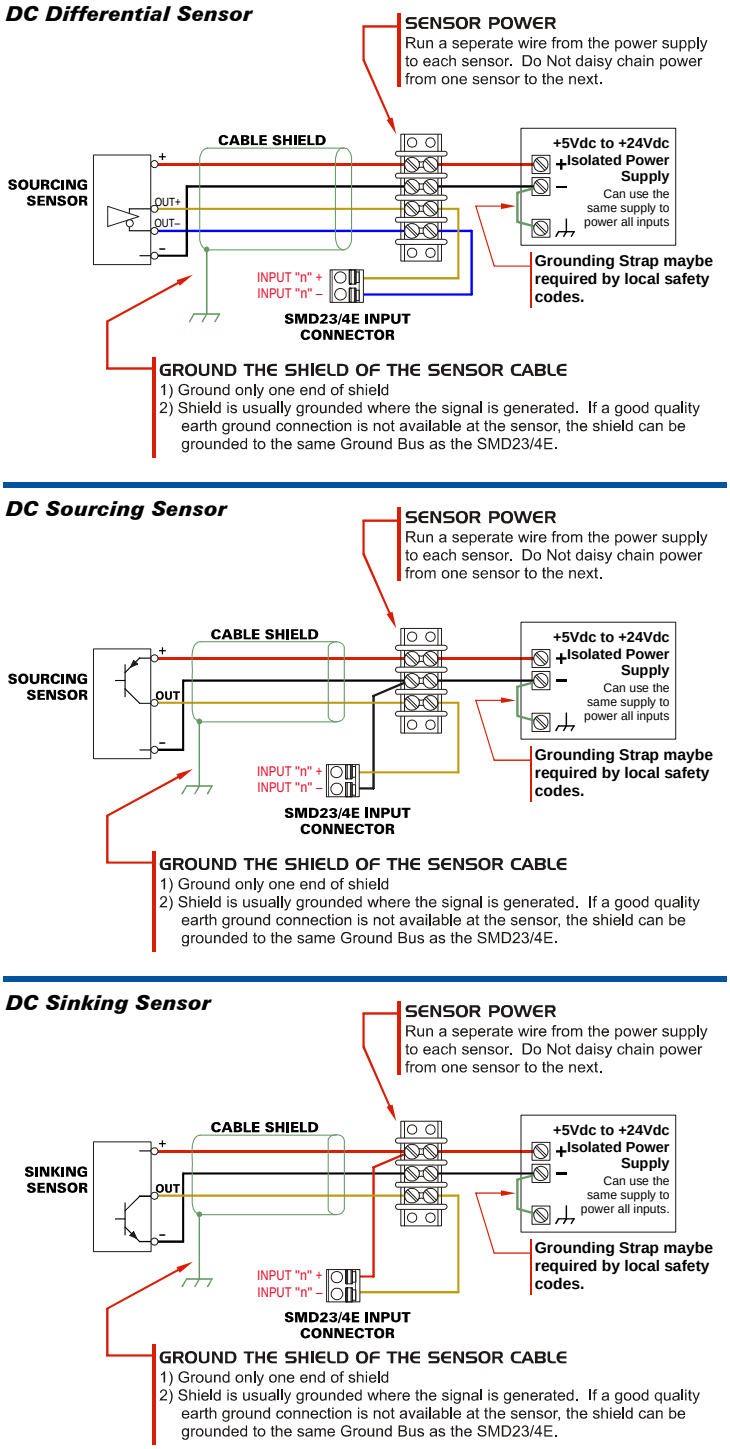

DC Differential Sensor
SENSOR POWER
Run a seperate wire from the power supply to each sensor. Do Not daisy chain power from one sensor to the next.
CABLE SHIELD
SOURCING SENSOR
OUT+
OUT–
+5Vdc to +24Vdc
+Isolated Power Supply
–
Can use the same supply to power all inputs
Grounding Strap maybe required by local safety codes.
INPUT "n" +
INPUT "n" –
SMD23/4E INPUT CONNECTOR
GROUND THE SHIELD OF THE SENSOR CABLE
1) Ground only one end of shield
2) Shield is usually grounded where the signal is generated. If a good quality earth ground connection is not available at the sensor, the shield can be grounded to the same Ground Bus as the SMD23/4E.
DC Sourcing Sensor
SENSOR POWER
Run a seperate wire from the power supply to each sensor. Do Not daisy chain power from one sensor to the next.
CABLE SHIELD
SOURCING SENSOR
OUT
+5Vdc to +24Vdc
+Isolated Power Supply
–
Can use the same supply to power all inputs
Grounding Strap maybe required by local safety codes.
INPUT "n" +
INPUT "n" –
SMD23/4E INPUT CONNECTOR
GROUND THE SHIELD OF THE SENSOR CABLE
1) Ground only one end of shield
2) Shield is usually grounded where the signal is generated. If a good quality earth ground connection is not available at the sensor, the shield can be grounded to the same Ground Bus as the SMD23/4E.
DC Sinking Sensor
SENSOR POWER
Run a seperate wire from the power supply to each sensor. Do Not daisy chain power from one sensor to the next.
CABLE SHIELD
SINKING SENSOR
OUT
+5Vdc to +24Vdc
+Isolated Power Supply
–
Can use the same supply to power all inputs.
Grounding Strap maybe required by local safety codes.
INPUT "n" +
INPUT "n" –
SMD23/4E INPUT CONNECTOR
GROUND THE SHIELD OF THE SENSOR CABLE
1) Ground only one end of shield
2) Shield is usually grounded where the signal is generated. If a good quality earth ground connection is not available at the sensor, the shield can be grounded to the same Ground Bus as the SMD23/4E.

Figure 4.5 Input Wiring

Input Specifications:

Differential. 2500 Vac/dc opto-isolated.† Can be wired as single ended inputs. Accepts 3.5 to 27Vdc without the need for an external current limiting resistor.

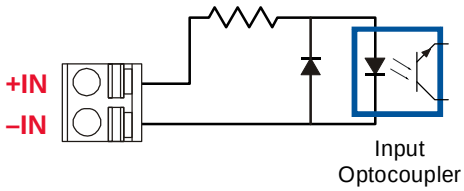


Figure 4.6 Simplified Input Schematic

Because they are low power signals, cabling from the sensor to the SMD23/4E should be done using a twisted pair cable with an overall shield. The shield should be grounded at the end when the signal is generated, which is the sensor end. If this is not practical, the shield should be grounded to the same ground bus as the SMD23/4E.

† This ratings refer to the optocouplers ability to prevent damaging voltages from crossing its isolation barrier to the rest of the SMD23/4E. If 120 Vac voltage is applied to an input, the input itself will still be destroyed.

Notes

CHAPTER 5
INSTALLING THE SMD23/4E-M12

This chapter applies to the SMD23/4E units that have M12 connectors. If you are installing a standard SMD23/4E unit, please refer to chapter 4 starting on page 45.

Location

SMD23/4E-M12 units are IP50 rated are suitable for use in an industrial environment that meet the following criteria:

- Only non-conductive pollutants normally exist in the environment, but an occasional temporary conductivity caused by condensation is expected.
- Transient voltages are controlled and do not exceed the impulse voltage capability of the product's insulation.

These criteria are equivalent to the *Pollution Degree 2* and *Over Voltage Category II* designations of the International Electrotechnical Commission (IEC).

SMD23/4E-M12P units are IP67 rated are suitable for use in an industrial environment that meet the following criteria:

- Continuous conductivity occurs due to conductive dust, rain, or other wet conditions.
- Transient voltages are controlled and do not exceed the impulse voltage capability of the product's insulation.

These criteria are equivalent to the *Pollution Degree 4* and *Over Voltage Category II* designations of the International Electrotechnical Commission (IEC).

Safe Handling Guidelines

Prevent Electrostatic Damage

CAUTION Electrostatic discharge can damage the SMD23/4E. Follow these guidelines when handling the unit.

1) Touch a grounded object to discharge static potential before handling the unit.
2) Work in a static-safe environment whenever possible.
3) Wear an approved wrist-strap grounding device.
4) Do not touch the pins of the network connector or I/O connector.
5) Do not disassemble the unit
6) Store the unit in its anti-static bag and shipping box when it is not in use.

Prevent Debris From Entering the Unit

WARNING While mounting of all devices, be sure that all debris (metal chips, wire strands, tapping liquids, etc.) is prevented from falling into the unit, specifically the M12 connectors. Debris may cause damage to the unit or unintended machine operation with possible personal injury.

Remove Power Before Servicing in a Hazardous Environment

WARNING Remove power before removing or installing any SMD23/4E units in a hazardous environment.

Mounting

All AMCI motor have flanges on the front of the motor for mounting. This flange also acts as a heatsink, so motors should be mounted on a large, unpainted metal surface. Mounting a motor in this fashion will allow a significant amount of heat to be dissipated away from the motor, which will increase the unit's life by reducing its operating temperature. If you cannot mount the motor on a large metal surface, you may need to install a fan to force cooling air over the SMD23/4E.

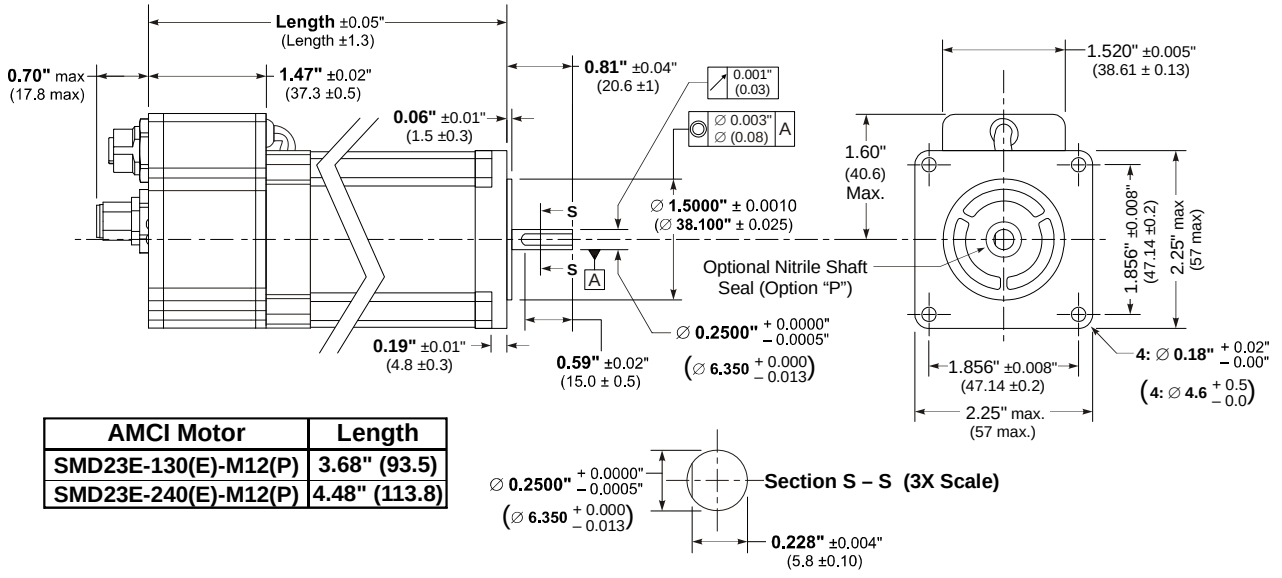
Motors should be mounted using the heaviest hardware possible. AMCI motors can produce high torques and accelerations that may weaken and shear inadequate mounting hardware.

NOTE

1) The motor case must be grounded for proper operation. This is usually accomplished through its mounting hardware. If you suspect a problem with your installation, such as mounting the motor to a painted surface, then run a bonding wire from the motor to a solid earth ground point near it. Use a minimum #8 gauge stranded wire or 1/2" wire braid as the grounding wire

2) Do not disassemble *any* stepper motor. A significant reduction in motor performance will result.

SMD23E-M12 Outline Drawing



AMCI Motor	Length
SMD23E-130(E)-M12(P)	3.68" (93.5)
SMD23E-240(E)-M12(P)	4.48" (113.8)

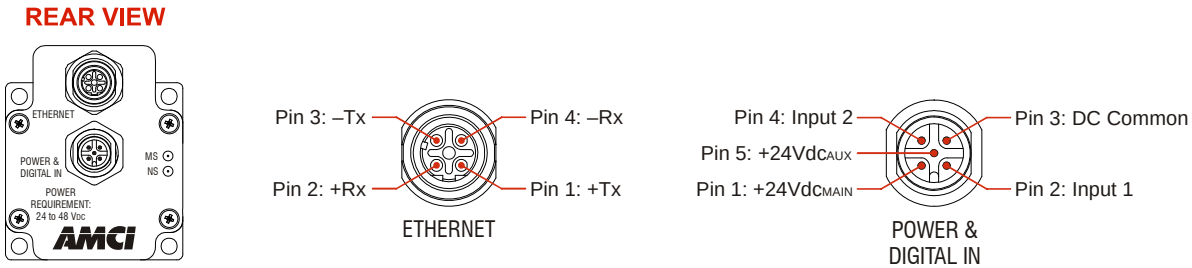


Figure 5.1 SMD23E-M12 Outline Drawing

Mounting (continued)

SMD24E-M12 Outline Drawing

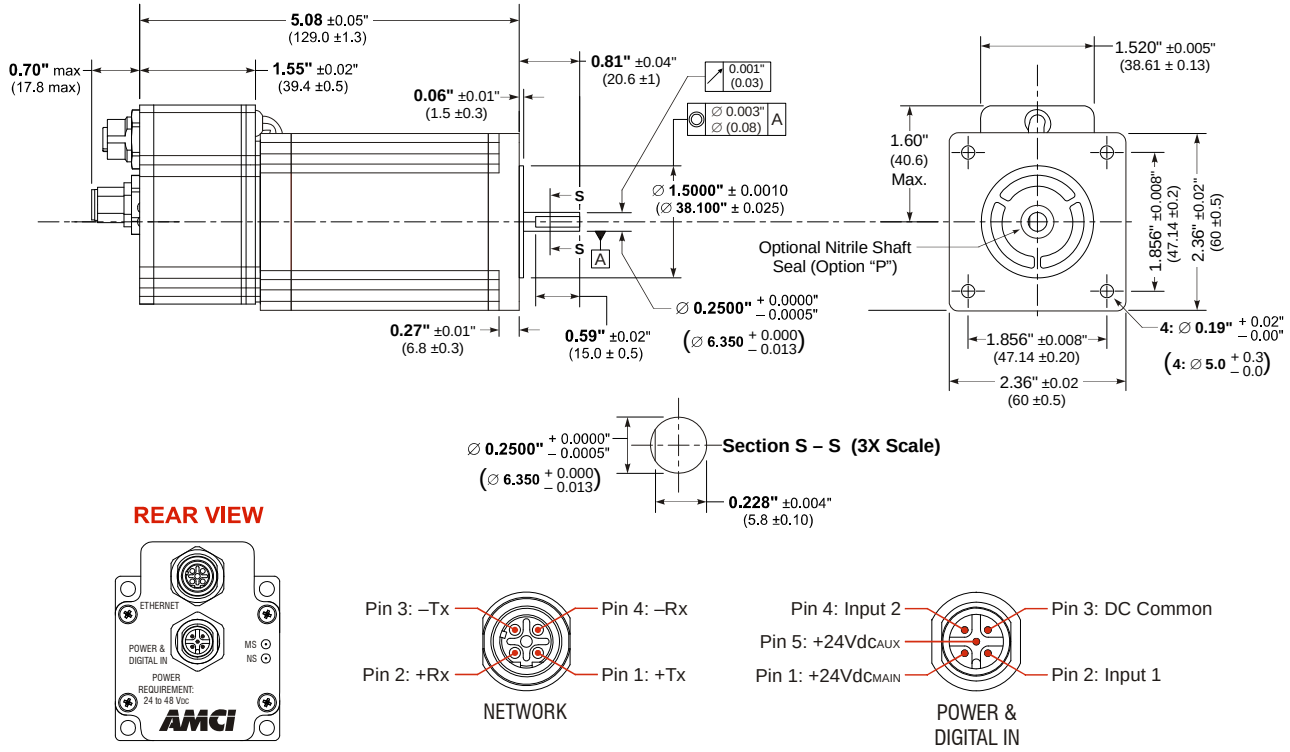


Figure 5.2 SMD24E-M12 Outline Drawing

SMD23/4E-M12 Mounting

The SMD23/4E-M12 is not water tight. Its IP50 rating makes it acceptable for use in dusty environments with occasional condensation. The SMD23/4E should be mounted in such a way that water will naturally drain off of the unit instead of pooling at the motor shaft, where the motor wires exit the motor, or on the motor laminations.

SMD23/4E-M12P Mounting

The SMD23/4E-M12P is water tight. Its IP67 rating makes it acceptable for use in washdown environments with a directed hose or in environments where it may be exposed to short term immersion in liquids.

Connecting the Load

Care must be exercised when connecting your load to the stepper motor. Even small shaft misalignments can cause large loading effects on the bearings of the motor and load. The use of a flexible coupler is *strongly* recommended whenever possible.

Network Connector

Figure 5.3 shows the Ethernet connector pinout when viewed from the back of the SMD23/4E-M12. The Ethernet port on the unit is an "auto-sense" port that will automatically switch between 10baseT and 100baseT depending on the network equipment it is attached to. The port also has "auto switch" capability. This means that a standard cable can be used when connecting the SMD23/4E to any device, including a personal computer.

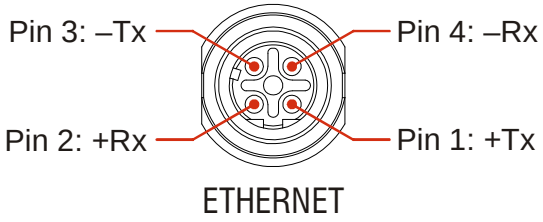


Figure 5.3 M12 Ethernet Connector Pinout

The connector is a standard female four pin D-coded M12 connector that is rated to IP67 when the mate is properly attached.

Compatible Connectors and Cordsets

Many different connectors and cordsets are available on the market, all of which will work with the SMD23/4E-M12 provided that the manufacturer follows the connector and Ethernet standards. AMCI has reviewed the following connectors and Ethernet cordsets for compatibility with the SMD23/4E-M12.

Connectors

AMCI #	Binder #	Description
MS-28	99-3729-810-04	Mating connector for Ethernet Connector. Male, four pin D-coded. Screw terminal connections. 6 to 8 mm dia. cable. Straight, IP67 rated when properly installed.

Table 5.1 Compatible Ethernet Connectors

Ethernet Cordsets

Phoenix Contact #	Description
1569391 SAC-4P-M12MSD/ 2,0-931	Bus system cable: 4-position, 26AWG, shielded. EIA/TIA 568B color coded Plug: Straight M12, D-coded, free conductor end. Cable length: 2 m
1569401 SAC-4P-M12MSD/ 5,0-931	Bus system cable: 4-position, 26AWG, shielded. EIA/TIA 568B color coded Plug: Straight M12, D-coded, free conductor end. Cable length: 5 m
1569414 SAC-4P-M12MSD/10,0-931	Bus system cable: 4-position, 26AWG, shielded. EIA/TIA 568B color coded Plug: Straight M12, D-coded, free conductor end. Cable length: 10 m
1569427 SAC-4P-M12MSD/15,0-931	Bus system cable: 4-position, 26AWG, shielded. EIA/TIA 568B color coded Plug: Straight M12, D-coded, free conductor end. Cable length: 15 m

Table 5.2 Ethernet Cordsets

NOTE These cordsets include the M12 connector, but the other end is un-terminated. This end can be punched down onto a patch panel, or an RJ45 connector can be added if the cordset is plugged directly into a switch.

Network Connector (continued)

TIA/EIA-568 Color Codes

There are two color codes in common use when wiring Ethernet connections with twisted pairs. Either one of these standards is acceptable. The Phoenix Contact cordsets that are suggested by AMCI all follow the 568B standard. Note that accidently reversing the Tx/Rx pairs will not affect the operation of the SMD23/4E. The SMD23/4E has an "auto-sense" port that will automatically adjust for swapped pairs.

Signal	568A Color	568B Color
+Transmit (+Tx)	White/Green Tracer	White/Orange tracer
–Transmit (–Tx)	Solid Green	Solid Orange
+Receive (+Rx)	White/Orange Tracer	White/Green Tracer
–Receive (–Rx)	Solid Orange	Solid Green

Table 5.3 TIA/EIA Color Codes

Input Connector

The Input Connector is located on the back of the SMD23/4E-M12 below the Ethernet Connector. Figure 5.4 shows the pin out for the Input Connector when viewed from the back of the unit.

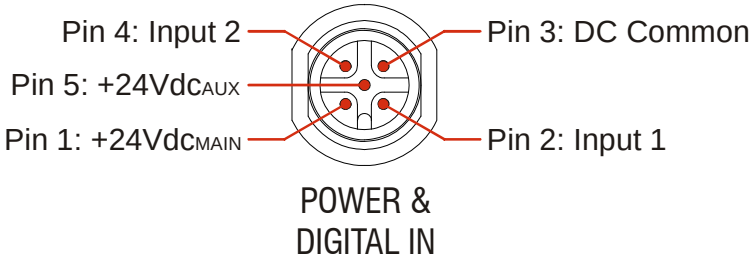


Figure 5.4 I/O Connector

Digital inputs on the SMD23/4E-M12 units are single ended and referenced to the DC Common pin. There are two power pins. $DCPower_{MAIN}$ powers both the control electronics and the motor. $DCPower_{AUX}$ powers only the control electronics. Using the $DCPower_{AUX}$ pin is optional. If your application requires you to cut power to your motor under some conditions, using the $DCPower_{AUX}$ pin allows you to cut power to your motor without losing your network connection. The connector is a standard five pin A-coded M12 connector that is rated to IP67 when the mate is properly attached.

Many different connectors and cordsets are available on the market, all of which will work with the SMD23/4E-M12 provided that the manufacturer follows the M12 A-coding standards. The following mating connector is available from AMCI.

AMCI #	Binder #	Description
MS-31	99-0436-12-05	Mating connector for Power Connector. Female, 5 pin A-coded. Screw terminal connections. 6 to 8 mm dia. cable. Straight, IP67 rated when properly installed.

Table 5.4 Compatible Input Connector

Power Wiring

The SMD23/4E accepts 24 to 48Vdc as its input power. AMCI strongly suggests using 18 AWG or larger wire for the power connections. The MS-31 connector will accept up to 18 gauge wire.

CAUTION Do not apply 120 Vac to any pins of the SMD23/4E. If this occurs, the unit will be damaged and you will void the unit's warranty.

Figure 5.5 below shows how to wire power to the SMD23/4E-M12 units. Note that Pin 5, $DCPower_{AUX}$, is only used when you introduce a circuit for removing power from the motor.

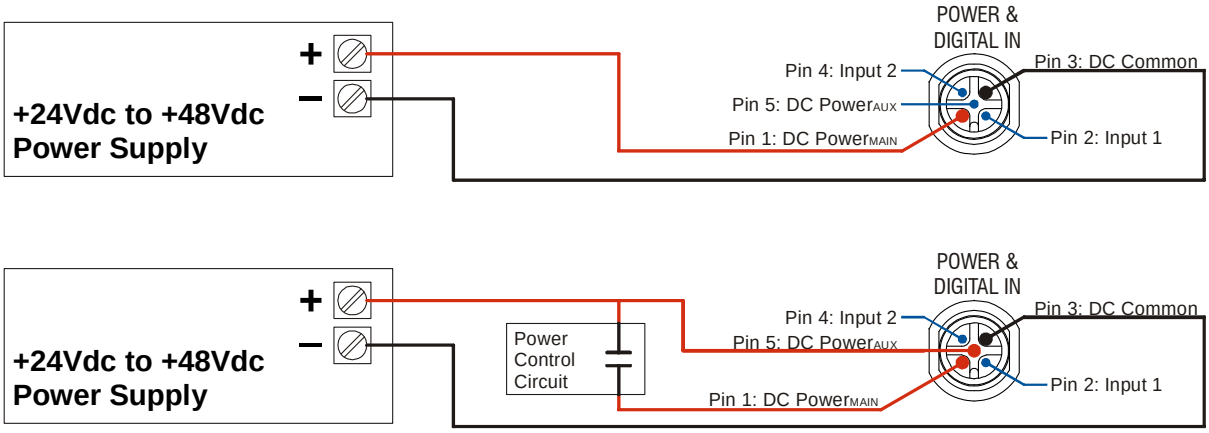


Figure 5.5 M12 Power Wiring

Input Wiring

Figure 5.6 below shows how to wire discrete DC differential, sourcing, and sinking sensors to Inputs 1 and 2 of the SMD23/4E-M12.

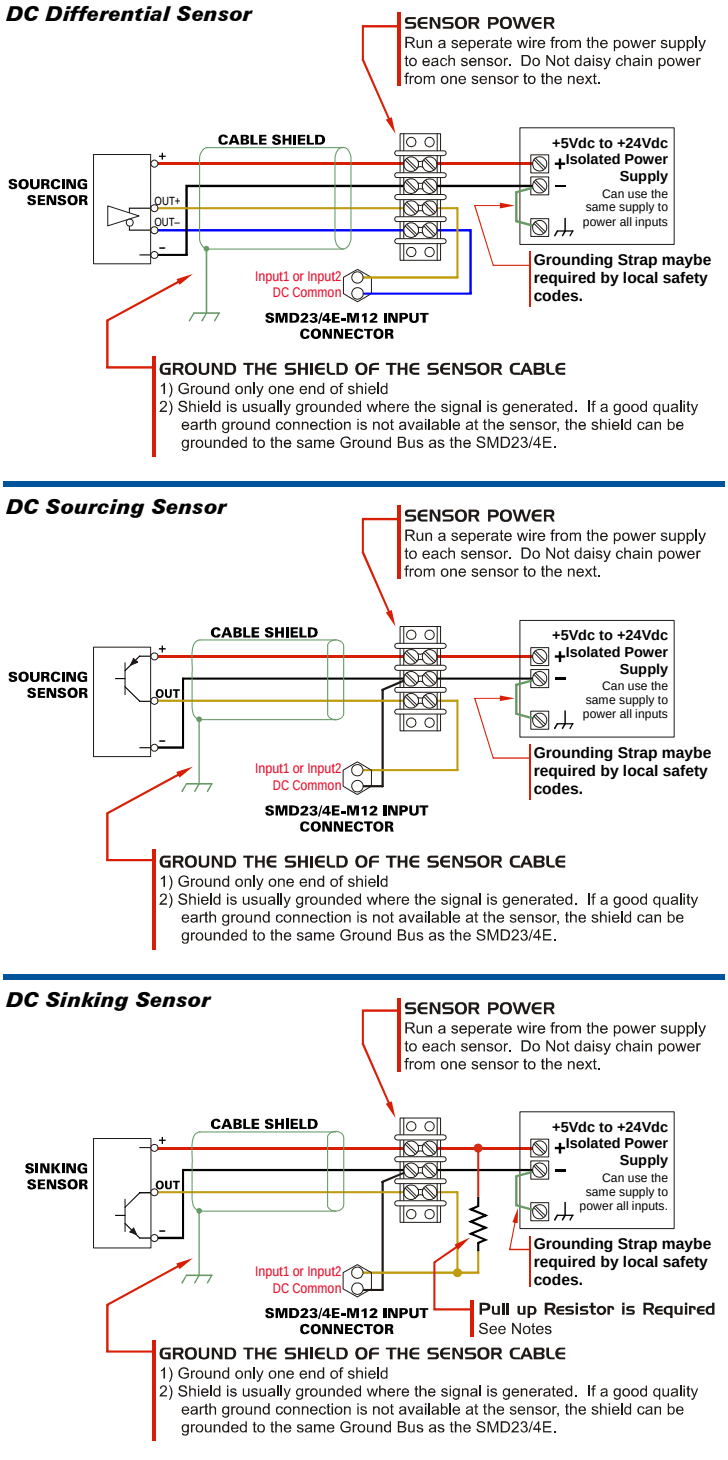

DC Differential Sensor
SENSOR POWER
Run a seperate wire from the power supply to each sensor. Do Not daisy chain power from one sensor to the next.
CABLE SHIELD
SOURCING SENSOR
OUT+
OUT–
+5Vdc to +24Vdc Isolated Power Supply
Can use the same supply to power all inputs
Grounding Strap maybe required by local safety codes.
Input1 or Input2
DC Common
SMD23/4E-M12 INPUT CONNECTOR
GROUND THE SHIELD OF THE SENSOR CABLE
1) Ground only one end of shield
2) Shield is usually grounded where the signal is generated. If a good quality earth ground connection is not available at the sensor, the shield can be grounded to the same Ground Bus as the SMD23/4E.
DC Sourcing Sensor
SENSOR POWER
Run a seperate wire from the power supply to each sensor. Do Not daisy chain power from one sensor to the next.
CABLE SHIELD
SOURCING SENSOR
OUT
+5Vdc to +24Vdc Isolated Power Supply
Can use the same supply to power all inputs
Grounding Strap maybe required by local safety codes.
Input1 or Input2
DC Common
SMD23/4E-M12 INPUT CONNECTOR
GROUND THE SHIELD OF THE SENSOR CABLE
1) Ground only one end of shield
2) Shield is usually grounded where the signal is generated. If a good quality earth ground connection is not available at the sensor, the shield can be grounded to the same Ground Bus as the SMD23/4E.
DC Sinking Sensor
SENSOR POWER
Run a seperate wire from the power supply to each sensor. Do Not daisy chain power from one sensor to the next.
CABLE SHIELD
SINKING SENSOR
OUT
+5Vdc to +24Vdc Isolated Power Supply
Can use the same supply to power all inputs.
Grounding Strap maybe required by local safety codes.
Input1 or Input2
DC Common
SMD23/4E-M12 INPUT CONNECTOR
Pull up Resistor is Required
See Notes
GROUND THE SHIELD OF THE SENSOR CABLE
1) Ground only one end of shield
2) Shield is usually grounded where the signal is generated. If a good quality earth ground connection is not available at the sensor, the shield can be grounded to the same Ground Bus as the SMD23/4E.

Figure 5.6 Input Wiring

Input Specifications:

Single Ended, 2500 Vac/dc opto-isolated.† Accepts 3.5 to 27Vdc without the need for an external current limiting resistor.

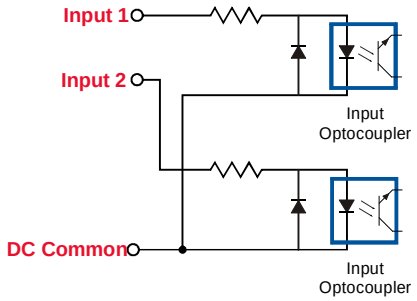


Figure 5.7 Simplified Input Schematic

Because they are low power signals, cabling from the sensor to the SMD23/4E should be done using a twisted pair cable with an overall shield. The shield should be grounded at the end when the signal is generated, which is the sensor end. If this is not practical, the shield should be grounded to the same ground bus as the SMD23/4E.

Sinking Output Sensors Require a Pull Up Resistor

Sinking output sensors require an external pull up resistor because the input to the SMD23/4E-M12 also sinks current. Table 5.5 below shows the values of pull up resistors that will allow the SMD23/4E input to activate along with the current that the sensor must be able to sink when it is active.

Input Voltage	Pull Up Resistor	Sensor Current When Active
5	300	16.7mA
12	1.4K	8.6mA
24	3.8K	6.3mA

Table 5.5 Pull Up Resistor

The logical states of the sensor and SMD23/4E input will be reversed. The SMD23/4E input is off when the sensor is active. You can set the logic state of the SMD23/4E input when you configure the unit.

† This ratings refer to the optocouplers ability to prevent damaging voltages from crossing its isolation barrier to the rest of the SMD23/4E. If 120 Vac voltage is applied to an input, the input itself will still be destroyed.

Notes

CHAPTER 6

NETWORK CONFIGURATION

This chapter covers how to configure the IP address of an SMD23/4E and set the communications protocol by using the AMCI Ethernet Configurator software. It is also possible to use a BOOTP server to set the IP address of an SMD23/4E. Instructions on using the BOOTP server from Rockwell Automation are available in Appendix A.

Factory Default

Each SMD23/4E ships from the factory with a default IP address of 192.168.0.50 and a subnet mask of 255.255.255.0. An SMD23/4E retains its IP address when power is removed. Therefore, the address of your SMD23/4E may be different if your unit is not directly from the factory.

Assumptions

This chapter makes a few assumptions:

1) You understand how IP addressing works and you only need instructions on setting the IP address of your SMD23/4E. If you do not yet understand IP addressing, there is plenty of information on the Internet. At the time of this writing, entering the phrase "ip addressing tutorial" into a popular search engine returned 93,000 hits.

2) You have administrative privileges on the computer you are using to configure the SMD23/4E. You will only need these privileges if you have to change the IP addressing of the network interface card used when configuring the unit.

3) You are configuring an SMD23/4E that has the factory default address of 192.168.0.50 and a subnet mask of 255.255.255.0. If you have an SMD23/4E that has already been configured with a different address, you must know this address and adjust these instructions accordingly. If you do not know the address of the SMD23/4E, the AMCI Ethernet Configurator software will still be able to connect to the module if they reside on the same subnet.

Firewalls

Firewalls are hardware devices or software that prevent unwanted network connections from occurring. Firewall software is present in Windows XP and above and it may prevent your computer for communicating with the SMD23/4E. If you think that your firewall may be interfering with the AMCI Ethernet Configurator software, then the easiest thing to do is temporarily disable the firewall from the Control Panel and enable it once you have finished configuring the SMD23/4E.

Installing the AMCI NET Software

Because many corporate firewalls prevent their users from downloading executable files, the install program exists as a ZIP file from our website. Installation is straight forward.

1) Download the software from ***http://www.amci.com/product-software.asp***. The link on the page is "AMCI configuration software for all networked products:..."

2) Extract the ZIP file to your computer. The resulting file is a simple install file.

3) Run the install program. When you do this, the following changes are made to your computer:

   - An AMCI folder is created in your programs folder. (Typically \Program Files\AMCI or \Program Files (x86)\AMCI.)
   - The AMCI NET Configurator program is placed in this directory
   - A shortcut to the AMCI NET Configurator program is created on the desktop and a program listing is created that will display the program off of the Start Menu.

NOTE No changes are made to the registry. The program can be easily un-installed by deleting the \Program Files\AMCI or \Program Files (x86)\AMCI directory and shortcuts.

Configure Your Network Interface Card (NIC)

Ideally, the NIC that you will use when configuring an SMD23/4E is not the NIC that attaches your computer to your corporate network. It is far easier to configure an SMD23/4E with a stand-alone laptop or when your computer has two NIC's in it, one for the corporate network and one for the SMD23/4E.

If you only have one NIC, and use it to attach your computer to your corporate network, you must disconnect your computer from the corporate network before configuring the SMD23/4E. DO NOT place a router between your computer and the corporate network so you can try to have access to the network and the SMD23/4E at the same time. The Network Administrator at AMCI can testify to the fact that "Bad Things" can happen when a rogue DHCP server in engineering starts to offer network information to computers around it. Save yourself and your IT department some headaches and disconnect yourself from your corporate network if you only have one NIC in your computer.

A wireless adapter does count as a network interface and if you have a laptop it may be possible to connect to your corporate network with the wireless interface and use a NIC to configure the SMD23/4E. It should also be possible to configure the SMD23/4E over a wireless connection by attaching it to a wireless router. However, this configuration has not yet been tested by AMCI.

Warning When Multiple NIC's Are On Your Computer

NOTE Having multiple NIC's enabled on your computer may cause connection issues. It appears to be caused by routing and default gateway settings on the computer. The easiest way to solve these problems is to disable all network interfaces except for the one used to attach to the SMD23E. (This includes wireless adapters.) If the SMD23E refuese to connect to your computer, or you see requests from the SMD23E but your computer does not respone correctly, disable all the additional network interfaces and try again.

The easiest way to check the current settings for your NIC is with the 'ipconfig' command.

- For Windows XP, click on the [Start] button, and click on [Run...]. A dialog box will open. Enter 'cmd' on the text line and press [Enter] on the keyboard.
- For Vista and Windows 7, click on the [Start] button, and type "cmd" in the "*Search programs and files*" text box. Press [Enter] on the keyboard.

A DOS like terminal will open. Type in 'ipconfig', press [Enter] on the keyboard and the computer will return the present Address, Subnet Mask, and Default Gateway for all of your network interfaces. If your present address is 192.168.0.x, where 'x' does not equal 50, and your subnet mask is 255.255.255.0, then you are ready to configure the SMD23/4E. Figure 6.1 shows the output of an ipconfig command that shows the "Local Area Connection 2" interface on the 192.168.0 subnet.

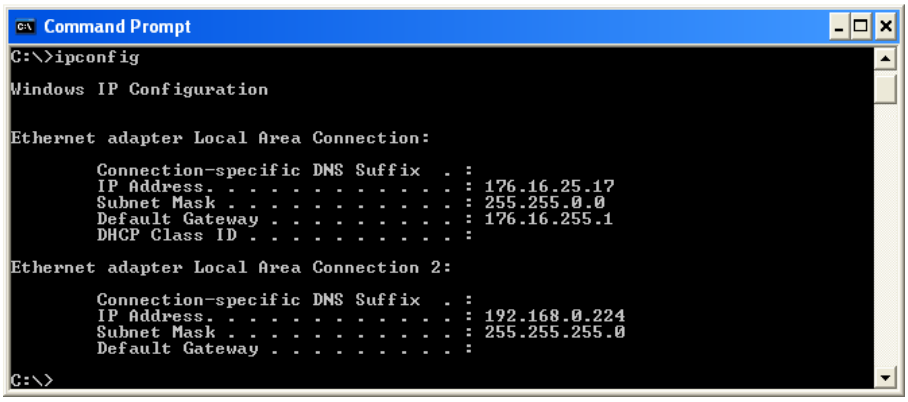

Command Prompt
C:\>ipconfig
Windows IP Configuration
Ethernet adapter Local Area Connection:
Connection-specific DNS Suffix . :
IP Address. . . . . : 176.16.25.17
Subnet Mask . . . . . : 255.255.0.0
Default Gateway . . . . . : 176.16.255.1
DHCP Class ID . . . . . :
Ethernet adapter Local Area Connection 2:
Connection-specific DNS Suffix . :
IP Address. . . . . : 192.168.0.224
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . :
C:\>

Figure 6.1 ipconfig Command

Configure Your Network Interface Card (NIC) (continued)

If your present address in not in the 192.168.0.x range, type in 'ncpa.cpl' at the command prompt and hit [Enter] on the keyboard.

- In XP, this opens the *Network and Dial-up Connections* window. Right click on the appropriate interface and select 'Properties' from the menu. This will open the *Properties* windows for the selected interface. Click on the 'Internet Protocol (TCP/IP)' component and then click on the [Properties] button.
- For Vista and Windows 7, this open the *Network Connections* window. Double click on the appropriate interface. In the window that opens, select "Internet Protocol Version 4 (TCP/IP v4)" from the list and then click on the [Properties] button.

Set the address, subnet mask to appropriate values. (192.168.0.1 and 255.255.255.0 will work for an SMD23/4E with factory default settings.) The default gateway and DNS server settings can be ignored.

Attach the SMD23/4E

The next step in configuring the SMD23/4E is attaching it to your computer. Any Cat5, 5e, or 6 cable can be used. You can attach the SMD23/4E directly to your computer or use a network switch. The SMD23/4E has an "auto-switch" port which eliminates the need for a crossover cable in direct connect applications.

Once the cables are attached, you can power up the SMD23/4E.

Using the AMCI Net Configurator

Start the Software

To start the program, click on the desktop icon or navigate from the start button to the program listing and click on the icon there. The program will present a screen similar to figure 6.2 below.

AMCI Net Configurator 5.08
Connect
Computer Comm. port
Ethernet Port
USB Port
Scan
Manual Connect
Connection Status: Unconnected
Device Identity
Name:
Vendor:
Product Type:
Product Code:
Major Rev.:
Minor Rev.
This software tool can be used to quickly setup and configure any AMCI networked product.
1. Connect your device to the computer through an Ethernet or USB port.
If needed, click on the [Help] button for additional instructions and cabling details
2. The AMCI device must not be connected to any other scanning device or controller.
3. Power your device. If this is the first time connecting an AMCI device through a USB port, your operating system may need to install device drivers.
4. Use the radio buttons to choose the appropriate connection type (Ethernet or USB)
5. With the Ethernet connection, you have two choices:
a. Click on the [Scan] button to see a list of available devices. Double click on the device you wish to attach to.
b. Click on the [Manual Connect] button. A text box will appear. Enter the IP address of the device you wish to attach to and press [Connect] button.
Note: The computer must be on the same subnet as the AMCI device.
6. With the USB connection, click on the [Connect] button to begin communicating with the device.
More detailed instructions are available by clicking the [Help] button.
AMCI ADVANCED MICRO CONTROLS INC.
EtherNet/IP
DeviceNet
CANopen
Modbus RTU
Modbus
PROFIBUS
PROFINET
AnyNet I/O
NEXUS I/O
NR25 Series Encoders
SMD Series Stepper
SD Series AC Stepper
Exit
Help

Figure 6.2 AMCI NET Configurator Opening Screen

Using the AMCI NET Software (continued)

Connect to your SMD23/4E

The first step in configuring your SMD23/4E is to connect to it. The software will attempt to find your unit if you click on the [Scan] button. If your unit is found, it will appear in the table below the [Scan] button. Simply double click on the IP address of the unit and the configurator screen will open.

The scan feature will not always work. This typically occurs when you have multiple network connections on your computer, (including wireless connections), and the SMD23/4E is not attached to the port configured as the default gateway. If the scan does not work, you can manually connect to your SMD23/4E if you know its IP address. Click on the [Manual Connect] button enter the IP address into the *Enter IP Address:* field and press [Connect]. (The factory default address of 192.168.0.50 is the default address of the field.)

Whether you scan or manually connect, you will end up at the screen shown in figure 6.3 below.

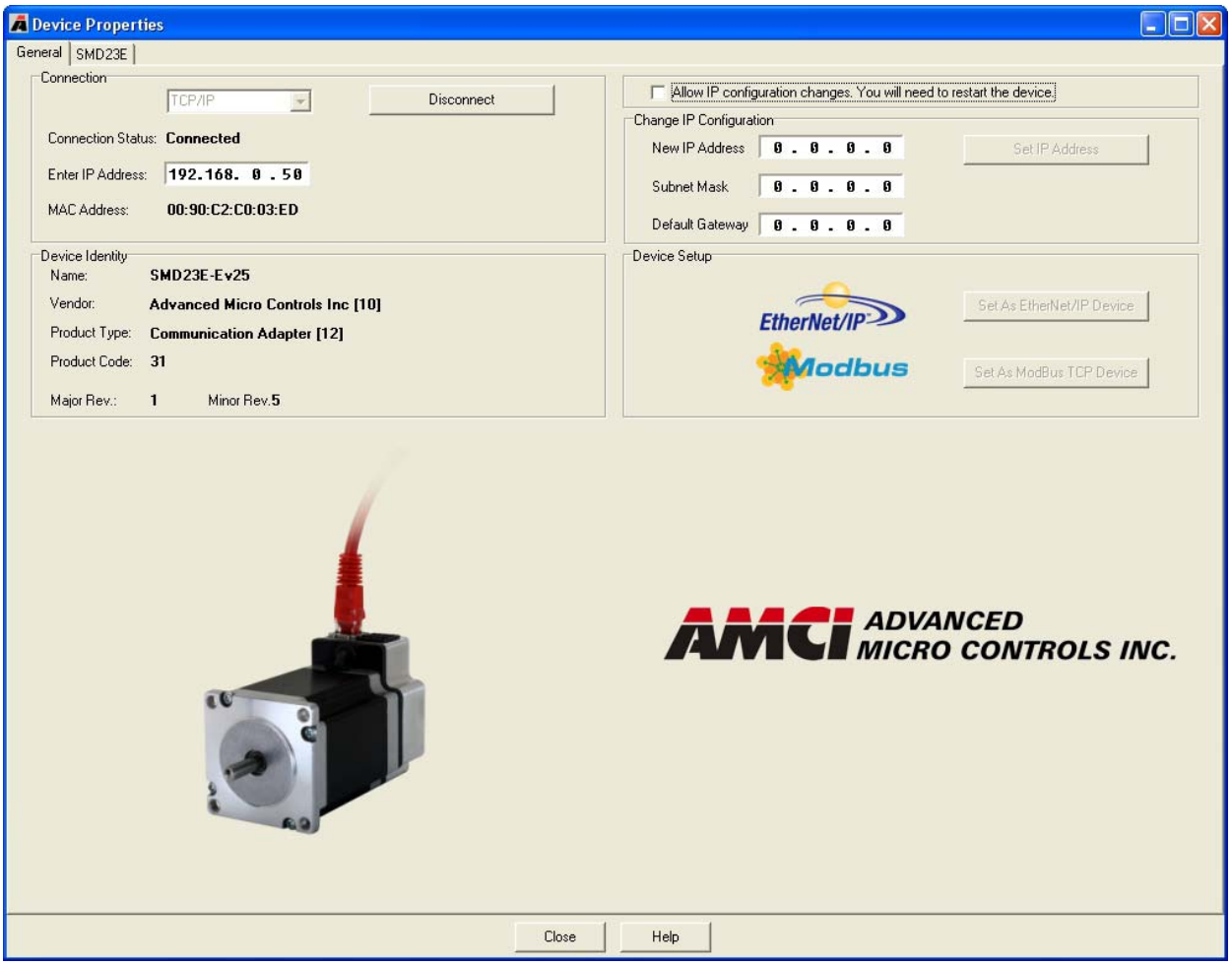


Figure 6.3 SMD23/4E Scan Results

You start with the General tab showing. The left half of this tab gives you information on the SMD23/4E that you are connected to. The right half of the window allows you to change the IP address and select the active protocol of the SMD23/4E. If you want to change this information, you must first allow edits by clicking on the check box at the top entitled “Allow IP configuration changes...”. The name of the SMD23/4E shows which protocol is active.

Using the AMCI NET Software (continued)

The Meaning of the Product Name

The Name field in on the left hand side of the General tab gives you clues to the configuration of the SMD23/4E as well as its firmware revision.

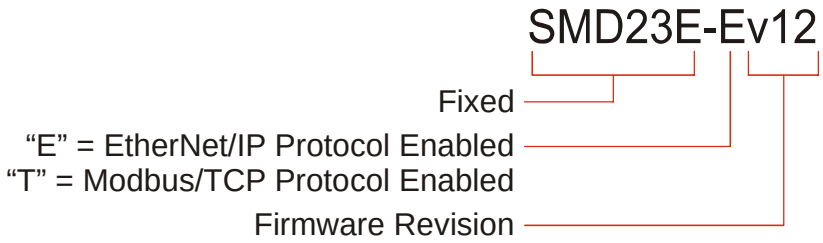


Figure 6.4 Product Name Description

Changing the IP Address

If you are changing the IP address, set the *New IP Address*, *Subnet Mask*, and *Default Gateway* fields to your desired values.

NOTE The *Default Gateway* must be set to a valid value, even if the SMD23E will never need to communicate with a different subnet. This means that the default gateway must be on the same subnet as the SMD23/4E but cannot be the address used by the SMD23/4E. AMCI suggests setting the default gateway to the address of your host controller.

Once these values are set, click on the [Set IP Address] button to write these values to the SMD23/4E. A pop up message will remind you that you must cycle power before the address change takes effect.

Setting Communications Protocol

Simply click on either of the buttons to change the communications protocol used by the SMD23/4E. When you click on either of the buttons, a pop up message will remind you that you must cycle power before the address change takes effect.

Cycle Power to the SMD23/4E

The SMD23/4E will not use the new configuration settings until the next time it powers up.

Testing the New Address

If you changed the address of the SMD23/4E to a new subnet, then change your NIC's address to be on the same subnet. Once this is done, apply power to the SMD23/4E. When the unit finishes its power up sequence press the [Scan] button and re-connect to the unit. Use this step will verify the new address and Network Protocol settings.

Using the AMCI NET Software (continued)

SMD23E Tab

When you connect to the SMD23/4E, a second tab is added to the window that allows you to read configuration data, and send configuration and command data to the unit.

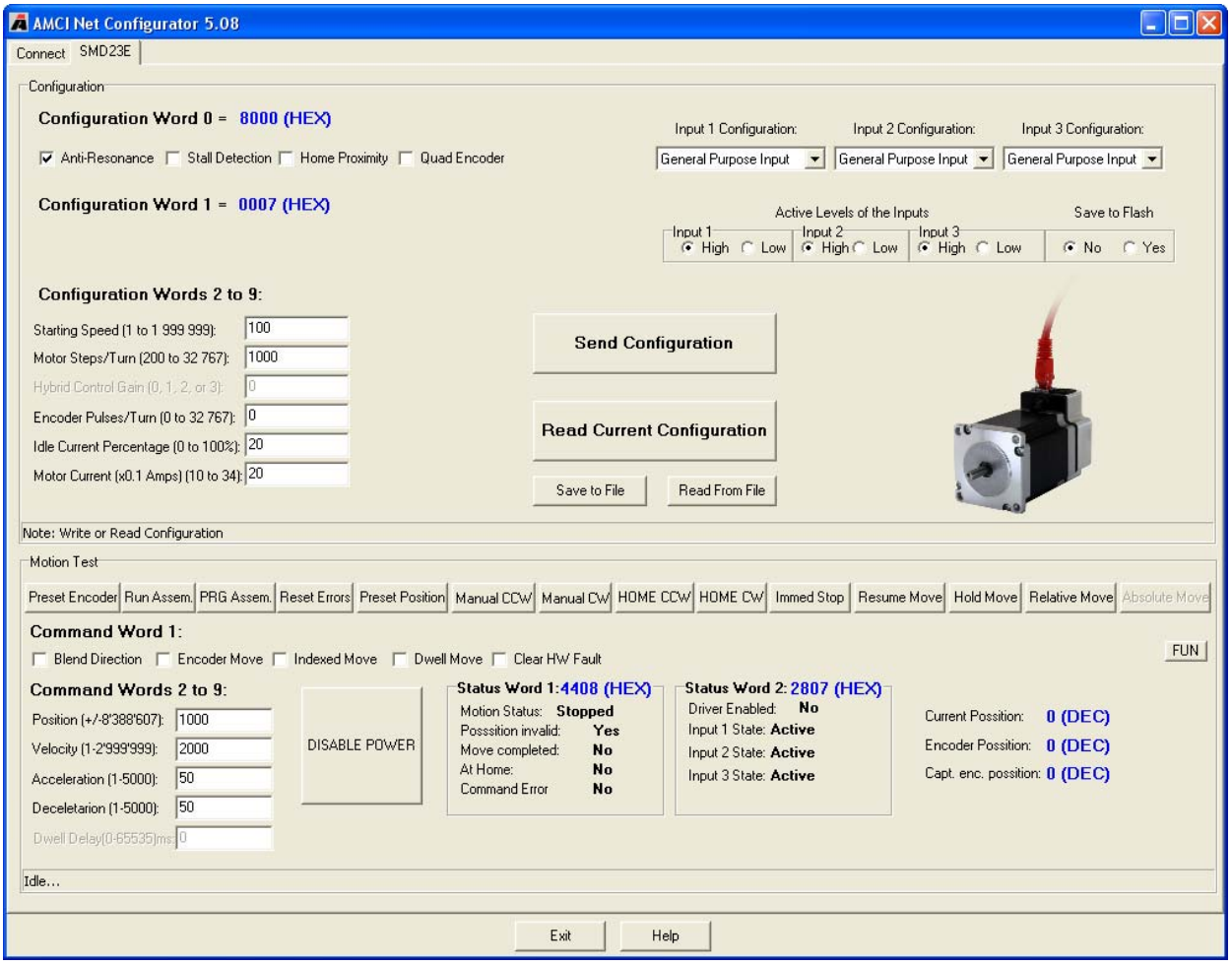


Figure 6.5 Motor Control Tab Screen

- The format of configuration and command data is covered in chapters 9 and 10 of this manual.
- The [Send Configuration] button will write the configuration data to the SMD23/4E. If you wish to store the configuration in Flash memory, first select the [Yes] radio button under the *Save to Flash* heading. If you choose to save the configuration to Flash, the unit will load the configuration on every power up and enter command mode. This eliminates the need to write configuration data to the SMD23/4E on every power up. However, AMCI sample programs assume the SMD23E will power up with a configuration error.
  - AMCI sample programs write down configuration values to the unit when it sees a configuration error. Using this functionality simplifies replacing an SMD23/4E if the need ever arises.

Using the Ping Command

The easiest way to test the address of the SMD23/4E when it is in your network is with the "ping" command. Before you can use the command, you have to be sure the SMD23/4E and your computer are still on the same subnet. For example, if the new address of the SMD23/4E is 192.168.0.42 and your computer has and address of 192.168.0.1, with a subnet mask of 255.255.255.0, then the two pieces of equipment are on the same subnet. (In this case, the first three numbers of the IP address must match.) If the new address of the SMD23/4E is 192.168.50.50, then the computer and SMD23/4E are not on the same subnet and you must go back into the Network Configuration panel and change your adapter's TCP/IP settings. Refer back to the ***Configure Your Network Interface Card (NIC)*** section of this chapter, starting on page 60.

When you are sure your computer and the SMD23/4E are on the same subnet, open the DOS terminal if necessary:

- For Windows XP, click on the [Start] button, and click on [Run...]. A dialog box will open. Enter 'cmd' on the text line and press [Enter] on the keyboard.
- For Vista and Windows 7, click on the [Start] button, and type "cmd" in the "*Search programs and files*" text box. Press [Enter] on the keyboard.

Once the terminal is open, type in 'ping aaa.bbb.ccc.ddd' where 'aaa.bbb.ccc.ddd' in the new IP address of the SMD23/4E. The computer will ping the SMD23/4E and the message "Reply from aaa.bbb.ccc.ddd: bytes=32 time<10ms TTL=128" should appear four times.

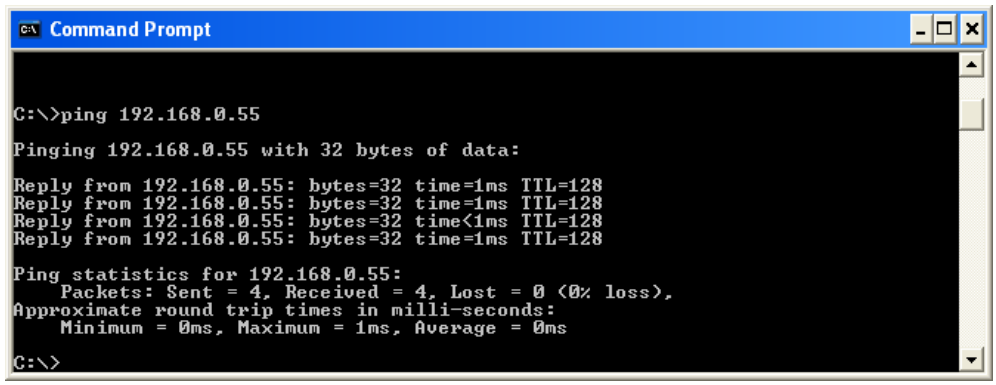

Command Prompt
C:\>ping 192.168.0.55
Pinging 192.168.0.55 with 32 bytes of data:
Reply from 192.168.0.55: bytes=32 time=1ms TTL=128
Reply from 192.168.0.55: bytes=32 time=1ms TTL=128
Reply from 192.168.0.55: bytes=32 time<1ms TTL=128
Reply from 192.168.0.55: bytes=32 time=1ms TTL=128
Ping statistics for 192.168.0.55:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>

Figure 6.6 IP Address Changed Successfully

If the message "Request timed out." or "Destination host unreachable" appears, then one of four things has occurred:

- You set the new IP address, but have not yet cycled power to the SMD23/4E
- You did not enter the correct address in the ping command.
- The new IP address of the SMD23/4E was not set correctly.
- The SMD23/4E and the computer are not on the same subnet.

Notes

CHAPTER 7

ETHERNET/IP CONFIGURATION

An SMD23/4E requires a host controller to issue configuration and motion commands to it. This chapter tells you how to configure Rockwell Automation ControlLogix, CompactLogix, and MicroLogix controllers to act as hosts for the SMD23/4E. All of these RA platforms use the EtherNet/IP communications protocol.

RSLogix 5000 Configuration

RSLogix 5000 is used to configure both the ControlLogix and CompactLogix platforms. When using these platforms, you have the option of using a separate Ethernet Bridge module or an Ethernet port built into the processor.

If the Ethernet port is built into processor, the only step you have to take before adding an SMD23/4E is to create a new project with the correct processor or modify an existing project. Once this is done, the Ethernet port will automatically appear in the Project Tree. If you are using a Ethernet Bridge module, you will have to add it to the I/O Configuration tree before adding the SMD23/4E to your project.

Configure Bridge Module (As Needed)

The first step is to create a new project or open an existing one. A 1756-L1 processor is used in the screen images below.

1) Insert a bridge module into the I/O Configuration tree. As shown in figure 7.1 on the right, right click on the I/O Configuration folder and select “New Module...” in the pop-up menu.

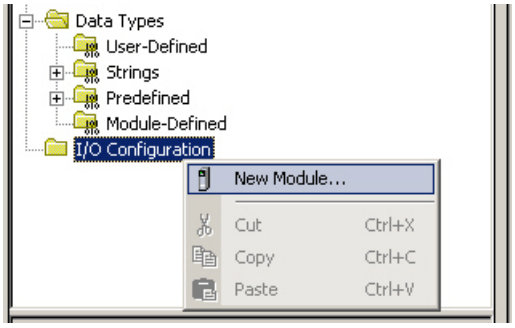


Figure 7.1 Defining a Bridge Module

RSLogix 5000 Configuration (continued)

Configure Bridge Module (As Needed) (continued)

2) In the *Select Module Type* windows that opens, select the proper Ethernet Bridge module. (In this example, the 1756-ENET/B.) Click on the [OK] button.

3) Enter the following information in the *Module Properties* window that opens. All parameters not listed here are optional. Figure 7.2 shows a completed screen.

- **Name:** A descriptive name for the Bridge Module.
- **IP Address:** Must be the address you want for the module, *not* the address you set for the SMD23/4E.
- **Slot:** The slot the module will reside in.

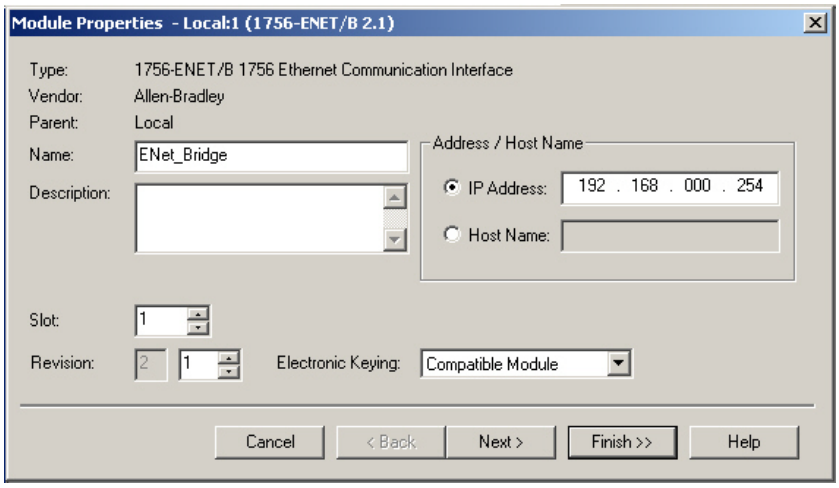


Figure 7.2 Setting Module Properties

4) When done, click on [Finish>>] to complete the setup of the Ethernet bridge module.

Configuring a Built-in Ethernet Port (As Needed)

You will still have to set an IP address for the Ethernet Port if the port is built into your processor. Right click on the port name in the I/O Configuration screen and select "Properties". A Module Properties window similar to the one shown in figure 7.2 will open. In this window you must set an IP Address for the port, *not* the IP address of the SMD23/4E.

RSLogix 5000 Configuration (continued)

Adding the SMD23/4E

You can add the SMD23/4E to the project once the Ethernet port (built-in or bridge module) is configured.

1) As shown in figure 7.3 on the right, the Ethernet port will be listed under the I/O Configuration tree. Right click on the port and then click on "New Module..." in the pop-up menu.
2) In the resulting window, click on the "By Vendor" tab towards the bottom of the window if necessary. Scroll down the list until you find the entry that has a Module Type of ETHERNET-MODULE and a description of "Generic Ethernet Module". Click on the module name to select and then click the [OK] button. A Module Properties window will open.
3) Set the following parameters in the Module Properties window. All parameters not listed here are optional. Figure 7.4 shows a completed screen.

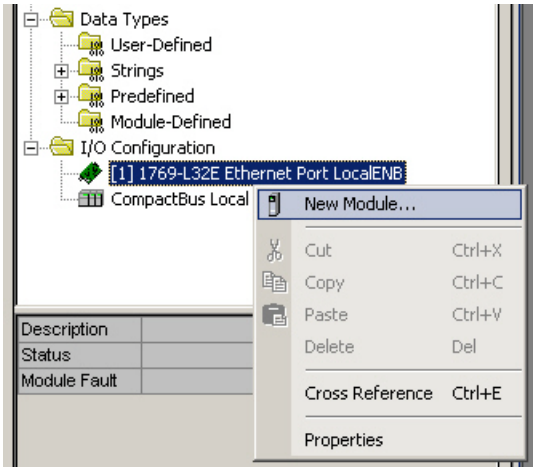


Figure 7.3 Adding an SMD23/4E

- **Name:** A descriptive name for the SMD23/4E.
- **Comm Format:** Data - INT ***(MUST be changed from the default Data - DINT.)***
- **IP Address:** Must be the address you set for the network module in the SMD23/4E. Refer to Chapter 6 starting on page 59 for information on setting the IP Address of an SMD23/4E.
- **Input:** Assembly Instance = 100, Size = 10 words.
- **Output:** Assembly Instance = 150, Size = 10 words.
- **Configuration:** Assembly Instance = 110, Size = 0

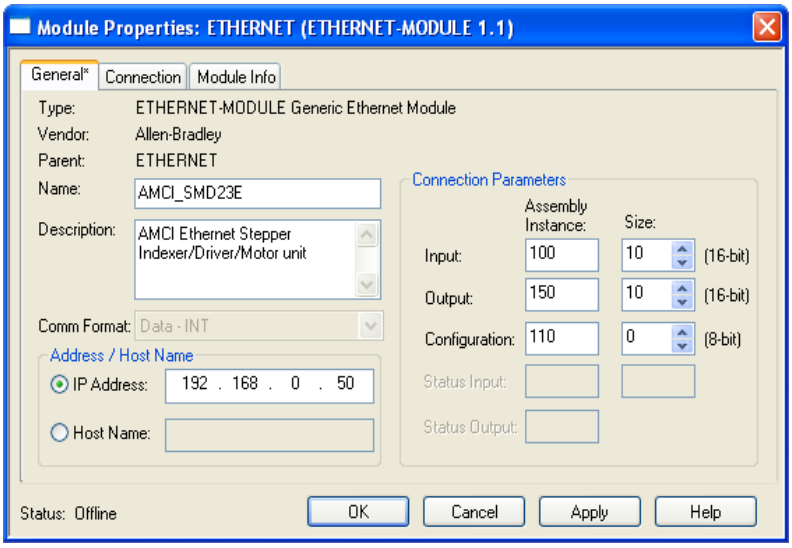


Figure 7.4 Sample SMD23/4E Configuration Screen

4) Click on [Next>]. Set the RPI time as required for your system. The minimum RPI time for an SMD23/4E is 1.5 milliseconds. When done, click on [Finish>>] to complete the setup.

RSLogix 500 Configuration

Platforms supported by the RSLogix 500 software package require Message Instructions to communicate with the SMD23/4E. This section uses a MicroLogix 1100 to describe how to configure these instructions.

Two instructions are required to transfer data between the PLC and the SMD23/4E. One instruction reads data from the unit and the other writes data to it. The following table gives the required attributes for the instructions.

	Read Instruction	Write Instruction
Service Type	Read Assembly	Write Assembly
Service Code	E (hex)	10 (hex)
Class	4 (hex)	4 (hex)
Instance	100 (decimal)	150 (decimal)
Attribute	3 (hex)	3 (hex)
Length	20 bytes	20 bytes

Table 7.1 Message Instruction Attributes

Using Message Instructions in a MicroLogix 1100 PLC

NOTE Only RSLogix 500 version 8.0 or above can be used to configure Message Instructions to communicate with an EtherNet/IP device.

1) Create four new data files.

- An Integer file to contain the data from the SMD23/4E. This file must be at least 10 words in length.
- An Integer file to contain the data sent to the SMD23/4E. This file must be large enough to contain all of the data written to the unit.
- A Message (MG) data file. This file must have at least two elements, one to control the Read Operation and one to control the Write Operation.
- An Extended Routing Information (RIX) data file. This file is used to store information used by the Message Instructions. This file must have at least two elements, one for the Read Operation and one for the Write Operation.

RSLogix 500 Configuration (continued)

Using Message Instructions in a MicroLogix 1100 PLC (continued)

2) Add the Message Instruction(s) to your Ladder Logic. The following rungs show how you can alternately read data from and write data to your SMD23/4E.

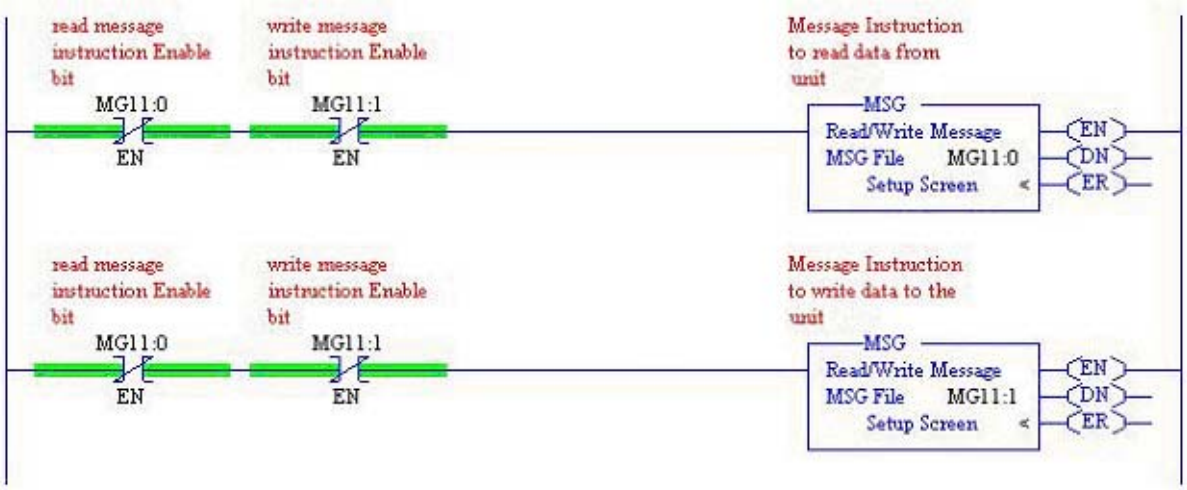


Figure 7.5 Message Instruction Example

3) Double Click on *Setup Screen* text inside the Message Instruction. The following window will open. Note that this is the default window and its appearance will change considerably as you progress through these steps.

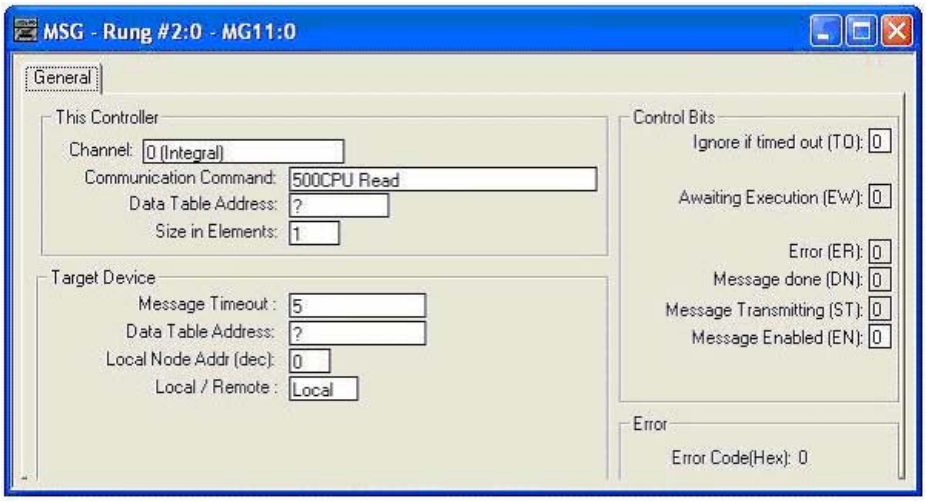


Figure 7.6 Message Instruction Setup Screen

4) Double click in the *Channel* field, click on the ▼, select "1 (Integral)", and press Enter.

5) Double click in the *Communication Command* field, click on the ▼, select "CIP Generic" and press Enter.

6) If the Message Instruction is being used to read data from the SMD23/4E, enter the integer file where the data will be placed in the *Data Table Address (Received)* field and press enter.

   If the Message Instruction is being used to write data to the SMD23/4E, enter the integer file where the source data will be located in the *Data Table Address (Send)* field and press Enter.

7) Enter "20" as the number of bytes needed in either the *Size In Bytes (Receive)* or *Size In Bytes (Send)* fields. Each SMD23/4E requires 20 bytes for both Receive and Send.

8) Enter a RIX address in the *Extended Routing* Info field. Please note that each Message Instruction must have its own RIX address.

RSLogix 500 Configuration (continued)

Using Message Instructions in a MicroLogix 1100 PLC (continued)

9) Double click in the *Service* field and select "Read Assembly" for a Message Instruction that is being used to read data from the SMD23/4E, or "Write Assemble" for a Message Instruction that is being used to send data to the SMD23/4E, and press Enter.

10) For *Read* operations, the *Service Code* field will change to "E" (hex). For *Write* operations, the *Service Code* field will change to "10" (hex). For both read and write operations, the *Class* field will change to "4" (hex), and the *Attribute* field will change to "3" (hex).

11) For Read operations, enter a value of 100 decimal (64 hex) in the *Instance* field.
For Write operations, enter a value of 150 decimal (96 hex) in the *Instance* field.

The figure below show a typical configuration for Message Instructions being used to read data from the SMD23/4E. Please note that the Data Table Address (Receive) field may be different in your application.

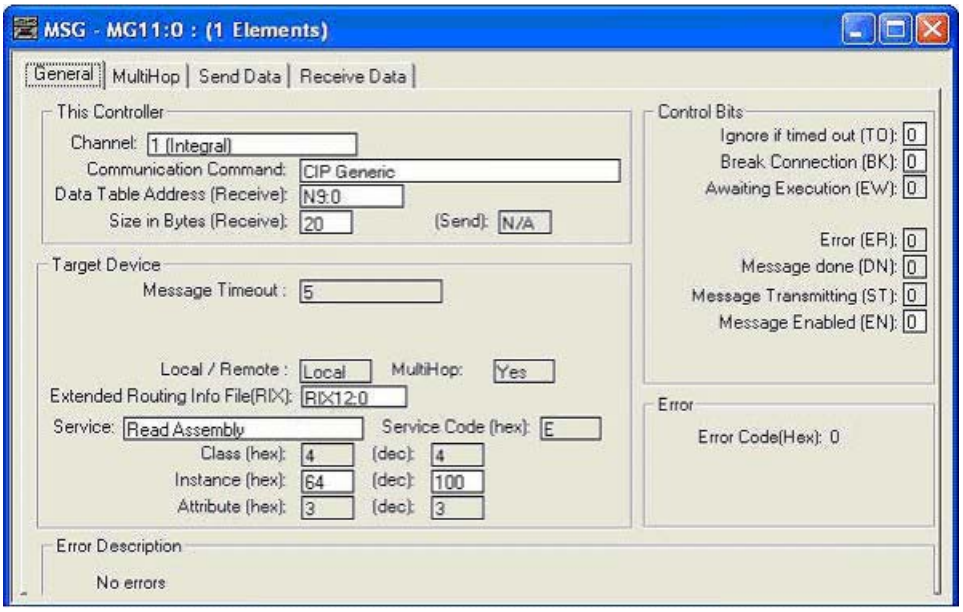


Figure 7.7 Read Message Instruction Setup Screen

RSLogix 500 Configuration (continued)

Using Message Instructions in a MicroLogix 1100 PLC (continued)

The figure below show a typical configuration for Message Instructions being used to write data to the SMD23/4E. Please note that the Data Table Address (Send) field may be different in your application.

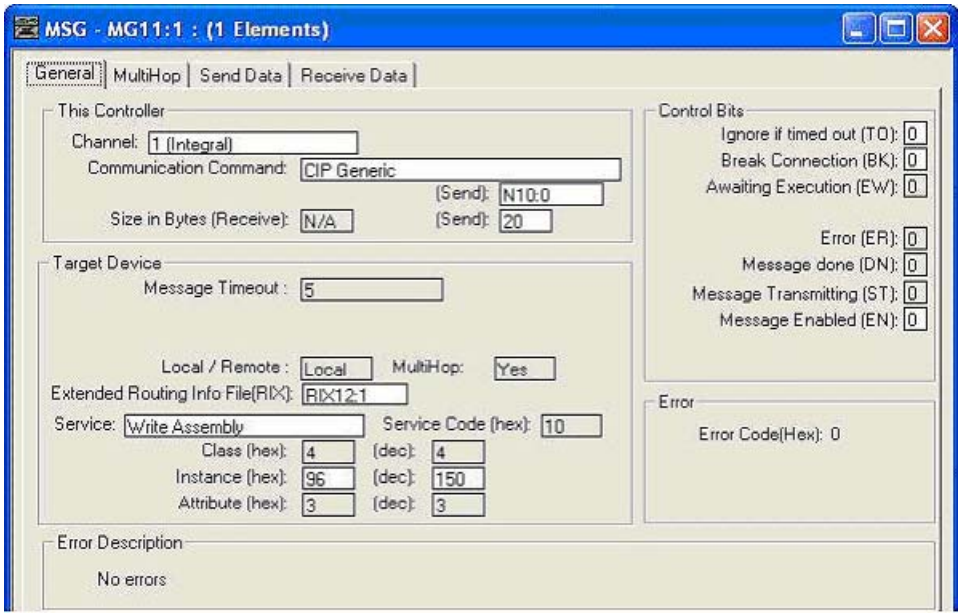


Figure 7.8 Write Message Instruction Setup Screen

Click on the MultiHop tab on the top of the window. As shown in figure 7.9, enter the IP address of the SMD23/4E and press Enter.

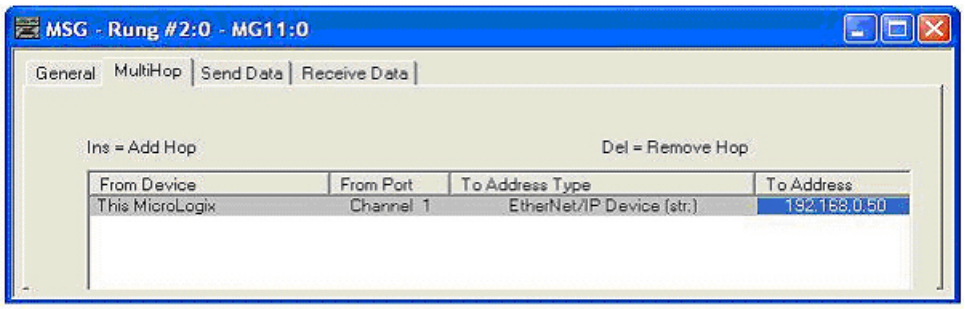


Figure 7.9 Message Instruction MultiHop Settings

After you are finished adding both the read and write message instructions to your program, save and download the program to the PLC.

RSLogix 500 Configuration (continued)

Using Message Instructions in a MicroLogix 1100 PLC (continued)

Troubleshooting

If you are unable to communicate with the SMD23/4E, the problem may be that the Ethernet port of your MicroLogix 1100 has not been configured. To check this, double click on Channel Configuration in the Project Tree and then select the Channel 1 tab. The following window will open.

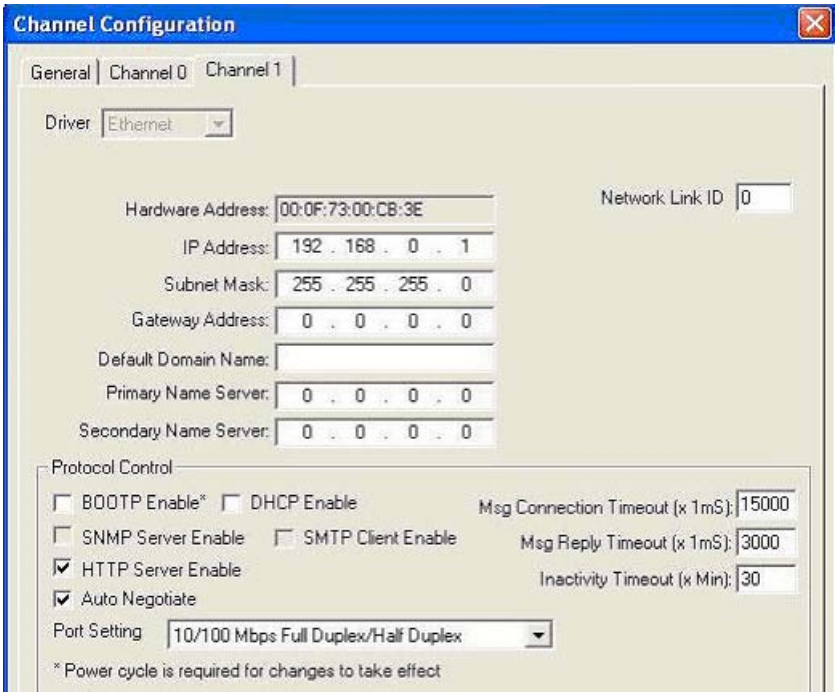


Figure 7.10 MicroLogix Ethernet Configuration Screen

Enter the IP address and Subnet Mask of your MicroLogix 1100, (not the address of the SMD23/4E) and click on [Apply]. The Ethernet Port should now be working.

CHAPTER 8
MODBUS TCP CONFIGURATION

An SMD23/4E requires a host controller to issue motion commands to these units. This chapter tell you how the I/O words used by an SMD23/4E are mapped to the Modbus I/O registers.

SMD23/4E Memory Layout

The SMD23/4E has a starting Input Register address of 0 and a starting Output Register address of 1024. Input Registers hold the data from the SMD23/4E while Output Registers hold the data to be written to the unit. Figure 8.1 shows how an SMD23/4E is mapped to the Modbus data reference. The complete specification for the Modbus protocol can be downloaded at http://www.modbus.org/specs.php.

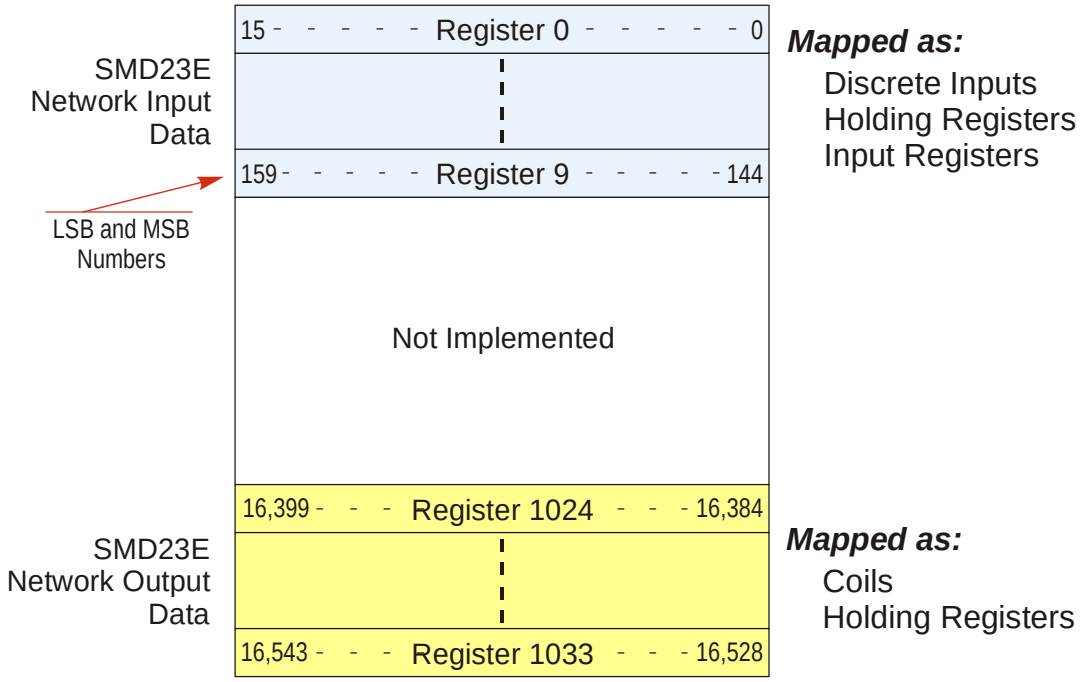


Figure 8.1 Modbus Data Reference Map: SMD23/4E

Supported Modbus Functions

Function Code	Function Name	SMD23/4E Register	Addressing method
1	Read Coils	OUTPUT	Bit: Addresses starting at 16,384
2	Read Discrete Inputs	INPUT	Bit: Addresses starting at 0
3	Read Holding Registers	OUTPUT & INPUT	Word: Out Regs. Starting at 1024 In Regs. Starting at 0
4	Read Input Registers	INPUT	Word: Addresses starting at 0.
5	Write Single Coil	OUTPUT	Bit: Addresses starting at 16,384
6	Write Single Register	OUTPUT	Word: Addresses starting at 1024
15	Write Multiple Coils	OUTPUT	Bit: Addresses starting at 16,384
16	Write Multiple Registers	OUTPUT	Word: Addresses starting at 1024
22	Mask Write Register	OUTPUT	Word: Addresses starting at 1024
23	Read/Write Registers	INPUT/OUTPUT	Word: Out Regs. Starting at 1024 In Regs. Starting at 0

Table 8.1 Supported Modbus Functions

Table 8.1 above lists all of the Modbus functions supported by the SMD23/4E units. AMCI supports all of these functions so that you can control the SMD23/4E as you see fit. However, if you are looking for the easiest way to interface with your unit, then you only need to use the *Read/Write Registers* function, which is function code 23.

NOTE Each SMD23/4E buffers the data that it sends over the network. If you use the *Read/Write Registers* function to write configuration data to the unit, then the data read with that command will not contain the response to the new configuration data. The response to the new data will be sent with the next data read.

Supported Modbus Exceptions

Code	Name	Description
01	Illegal function	The module does not support the function code in the query
02	Illegal data address	The data address received in the query is outside the initialized memory area
03	Illegal data value	The data in the request is illegal

Table 8.2 Supported Modbus Exceptions

Sample Modbus Configuration

The following section uses the Schneider-Electric Modicon M340 platform as an example of how to configure a Modbus TCP device. Unity Pro 6.0 software is used along with a BMXP342020 processor and a BMXNOC0401 Ethernet Communications module. The example assumes that you are starting a new project. You may have to adjust module location and data addressing to match your existing application.

Initial Configuration

From the menu bar, click on File -> New (Ctrl+N) and select your processor. (P342020 in this example.)

In Project browser, double click on the PLC bus. The Hardware catalog will open along with a picture of your rack.

In the Hardware catalog, expand the Communication section and drag the name of the communications module to the proper slot of the rack picture. (A BMX NOC 0401 in slot 1 is used in this example.) A "Properties of device" window will open that allows you to rename the device. As shown in figure 8.2, it is left to its default name in this example.

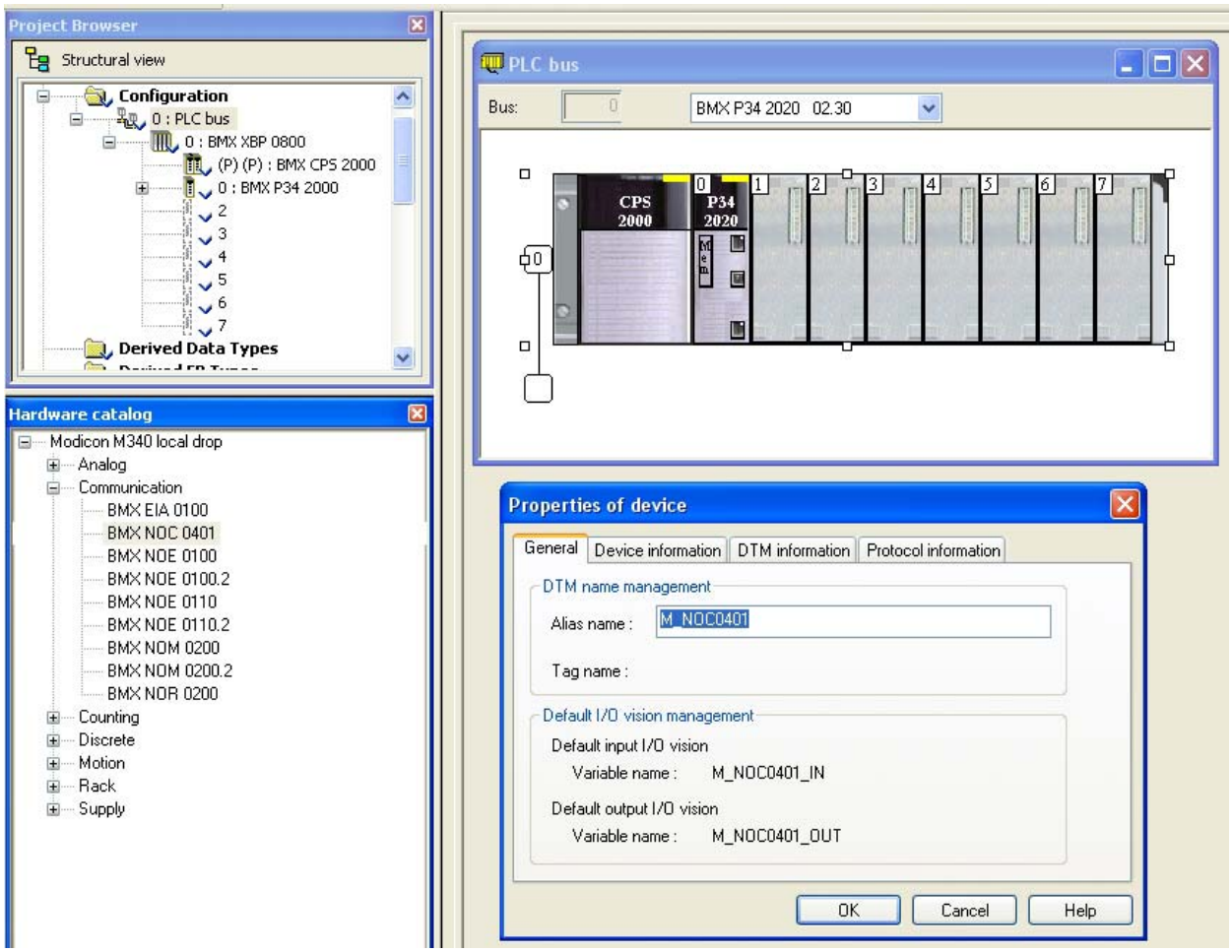


Figure 8.2 Configuring the M340 Rack

Once you set the name, click [OK] and the device is added. Add all other devices needed by your application and then close the PLC bus window to close the Hardware catalog.

Sample Modbus Configuration (continued)

Configuring the Ethernet Scanner

In the Project Browser, expand the rack if necessary and double click on the communications module. (The BMX NOC 0401 in this example.) This will open a window that allows you to set the number of I/O words assigned to the device.

NOTE The number of input and output words has to be large enough to accommodate all of the devices connected to the NOC0401 plus the sixteen input and sixteen output words needed by the NOC0401 for communication status bits. If a single SMD23/4E unit is attached to the NOC0401, you will need twenty-six I/O words.

As shown in figure 8.3, the %MW index values must be even.

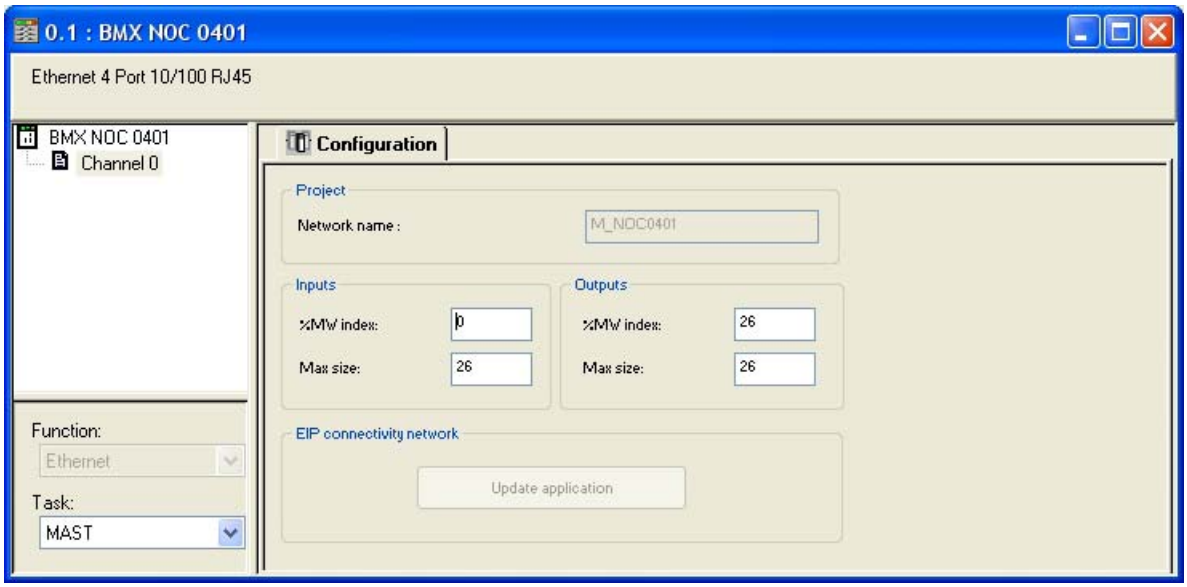


Figure 8.3 Assigning IO Words to Ethernet Module

Once you update the index and size values, click on Edit -> Validate (Ctrl+W) to validate the change. Then press the [Update application] button to create the Derived Data Types. You can then close the BMX NOC0401 window.

Adding the SMD23/4E

Click on Tools -> DTM Browser (Alt+Shift+1) to open the DTM Browser window. Double click on the NOC0401 module. In the "*fdtConfiguration*" window that opens, select *Channel Properties + TCP/IP* to set the Scanner IP Address, Sub-Network Mask, and Gateway IP Address. The Scanner IP address must be on the same subnet as the SMD23/4E. Close the "*fdtConfiguration*" window.

In the DTM Browser, right click on the NOC0401 and select "Add..." In the "*Add*" window that opens, scroll down to "Modbus Device" and double click on it. Enter an Alias name for the SMD23/4E. In this example, the name is set to "SMD23/4E_50". (50 is the last octet of its IP address) Click [OK] to close the window.

In the DTM Browser, double click on the NOC0401. A configuration window will open. Under "Device List" on the left side, click on the SMD23/4E_50 entry. There are three tabs that have information that must be set.

Sample Modbus Configuration (continued)

Adding the SMD23/4E (continued)

Under the *Properties* tab, change the "Import Mode:" to "Manual". See figure 8.4 below. Click on the [Apply] button.

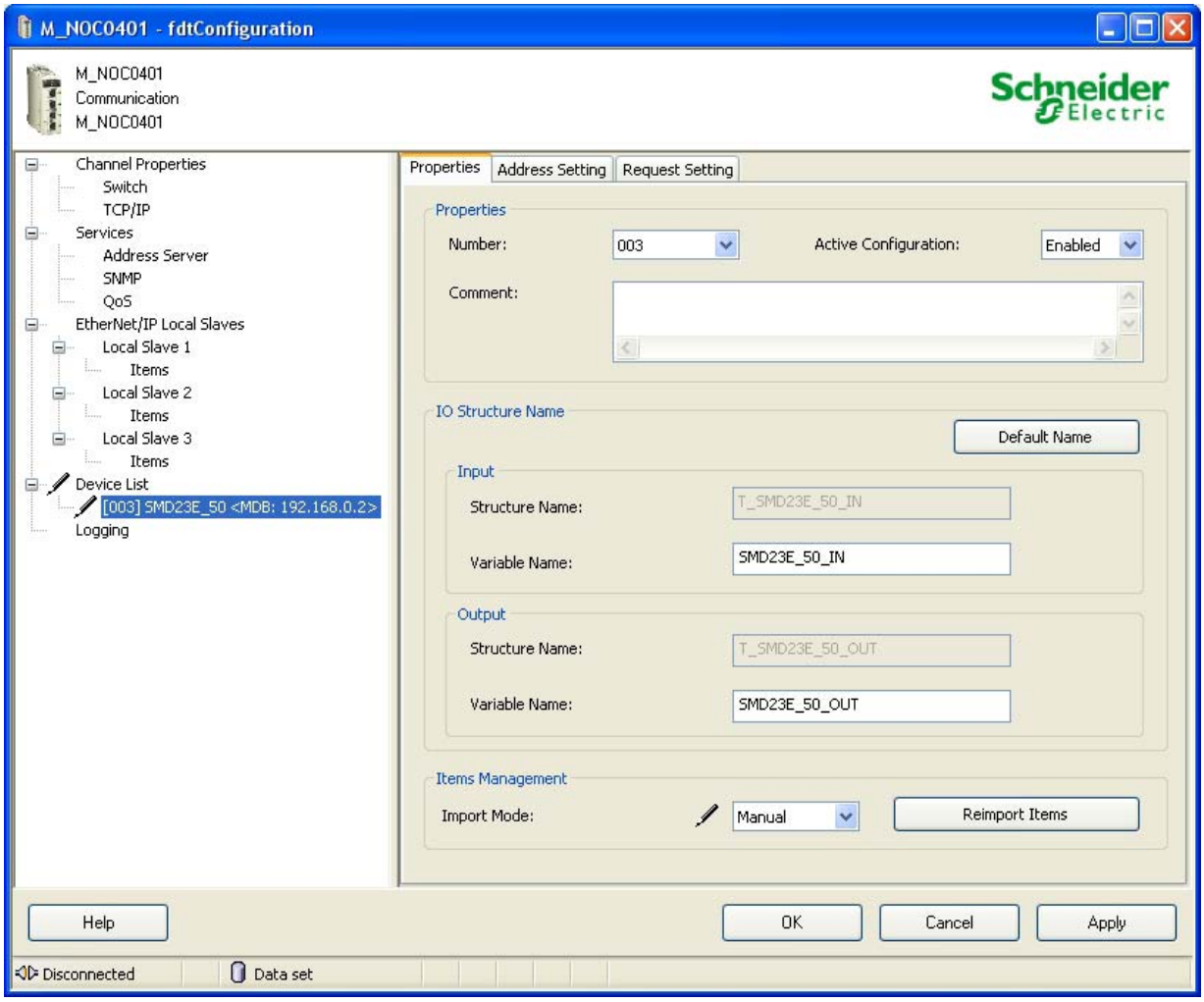


Figure 8.4 AnyNET-I/O Stack Properties

Sample Modbus Configuration (continued)

Adding the SMD23/4E (continued)

Under the *Address Setting* tab, set the IP Address to the one used by the SMD23/4E. SMD23/4E units do not use DHCP, so do not enable it for these devices. See figure 8.5 below. Press [Tab] to leave the IP address field and the click on the [Apply] button.

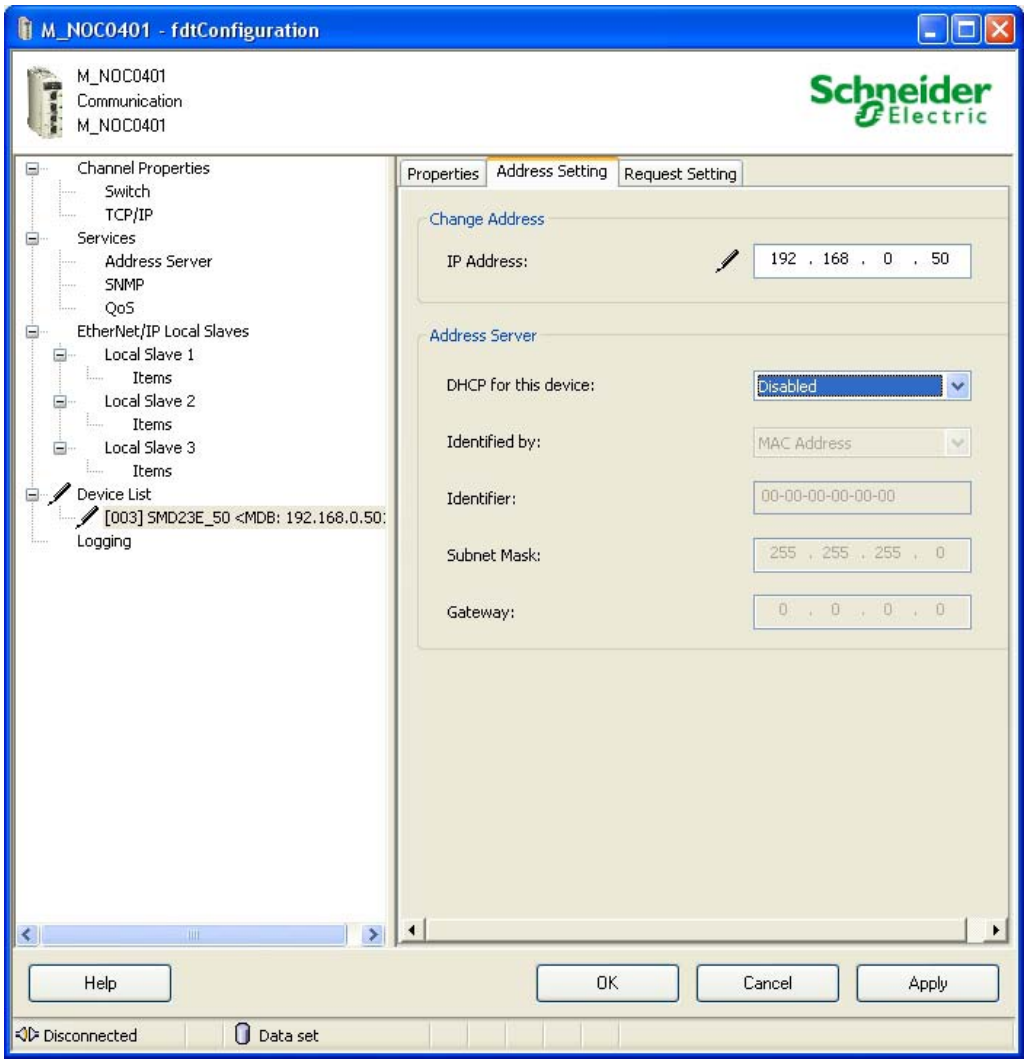


Figure 8.5 AnyNET-I/O Stack Addressing

Sample Modbus Configuration (continued)

Adding the SMD23/4E (continued)

The *Request Setting* tab is used to set up implicit communications between the PLC and the SMD23/4E. The scanner will automatically update the unit at the programmed rate if you configure these settings. It is also possible to communicate with the SMD23/4E under program control by addressing the unit in your code.

Click on the [Add Request] button to start the settings.

- **Connection Bit:** Leave as is
- **Unit ID:** Leave as is
- **Health Time Out (ms):** The PLC sets an error if the SMD23/4E does not respond in this amount of time. This setting must be greater that the Repetitive Rate setting. 1500 milliseconds has been used in testing at AMCI.
- **Repetitive Rate (ms):** How often the SMD23/4E is updated. Range of 5 to 60,000 milliseconds, with intervals of five milliseconds. An SMD23/4E can be updated every five milliseconds, but you may want to increase this time based on need and network load.
- **RD Address:** The starting address of the read data. The SMD23/4E read address starts at 0.
- **RD Length:** The number of words transferred by the request from the SMD23/4E. This should always be set to ten.
- **Last Value:** Choose what will happen to the data if communication is lost.
- **WR Address:** The starting address of the write data. The SMD23/4E write address starts at 1024.
- **WR Length:** The number of words transferred by the request to the SMD23/4E. This should always be set to ten.

Figure 8.6 shows the tab after the request is configured to read the SMD23/4E every five milliseconds

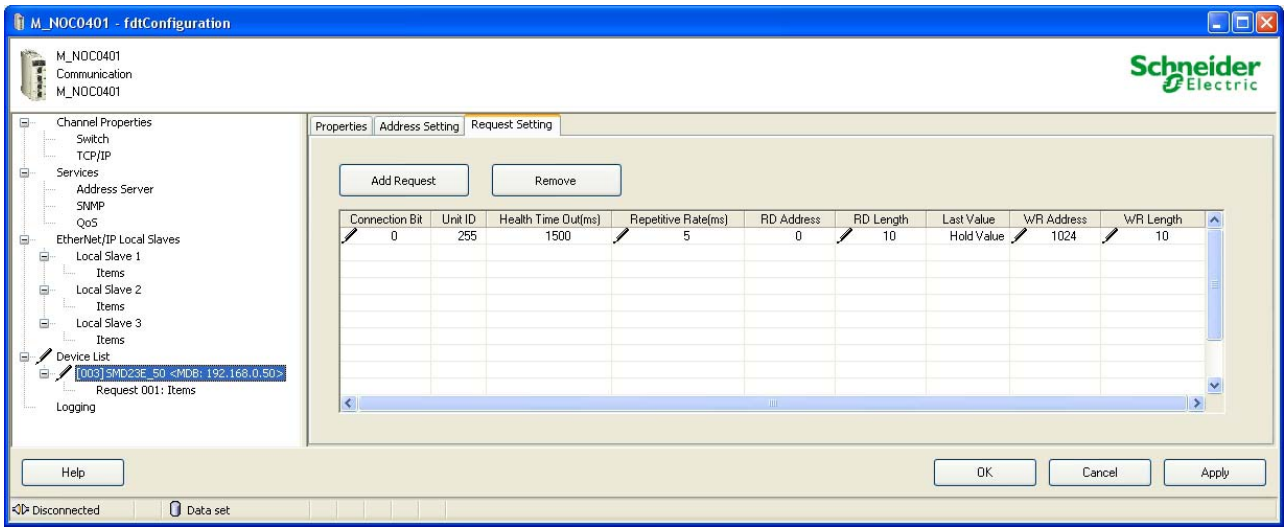


Figure 8.6 Configuring a Stack Transfer Request

Once you have set the parameters for the AnyNET-I/O stack, click the [OK] button to save the parameters and close the window.

Sample Modbus Configuration (continued)

Download Project to PLC

In the menu bar, click on Build -> Analyze Project. The test should complete without error and the Derived Data Types for the SMD23/4E should appear in the Project Browser.

In the menu bar, click on Build and either Build Changes or Rebuild All Project. This will compile the project.

With the PLC connected to your PC, and the SMD23/4E attached to the NOC0401, click on PLC -> Standard Mode to be sure you are going to communicate with the BMXP342020. Then click PLC -> Connect (Ctrl+K) to connect to the PLC.

After communication is established, click on PLC -> Transfer Project to PLC (Ctrl+L). In the window that opens, click the [Transfer] button. The project is downloaded to the PLC.

Click on PLC -> Run (Ctrl+R) to place the PLC in Run Mode.

View SMD23/4E Data

In order to see the values coming from the SMD23/4E, you must create an Animation Table. In the Project Browser, right click on the Animation Tables and select *New Animation Table*. Give the table an appropriate name, such as "SMD23/4E_50" which is used in this example. Unity Pro creates an empty table. Double click on the first cell of the table and then click on the ellipsis "..." button to open the "*Instance Selection*" window.

In the Instance Selection window, hold down the shift key and select the input and output words assigned to the SMD23/4E. Click the [OK] button to populate your Animation Table.

Sample Modbus Configuration (continued)

View SMD23/4E Data (continued)

Figure 8.7 on the following page shows a populated animation table for the input and output words assigned to the SMD23/4E. When using the Modbus Device DTM, the data is organized into bytes and the names are fixed. The M340 platform uses the little-endian format, so the least significant byte is in the even number word. For example, figure 8.7 shows the input data from an SMD23/4E. Free0[0] and Free0[1] make up Status Bits MSW for the unit, with Free0[0] the least significant byte. The combined value of 16#6408 is the value the SMD23/4E sends over the network on power up if it is not configured.

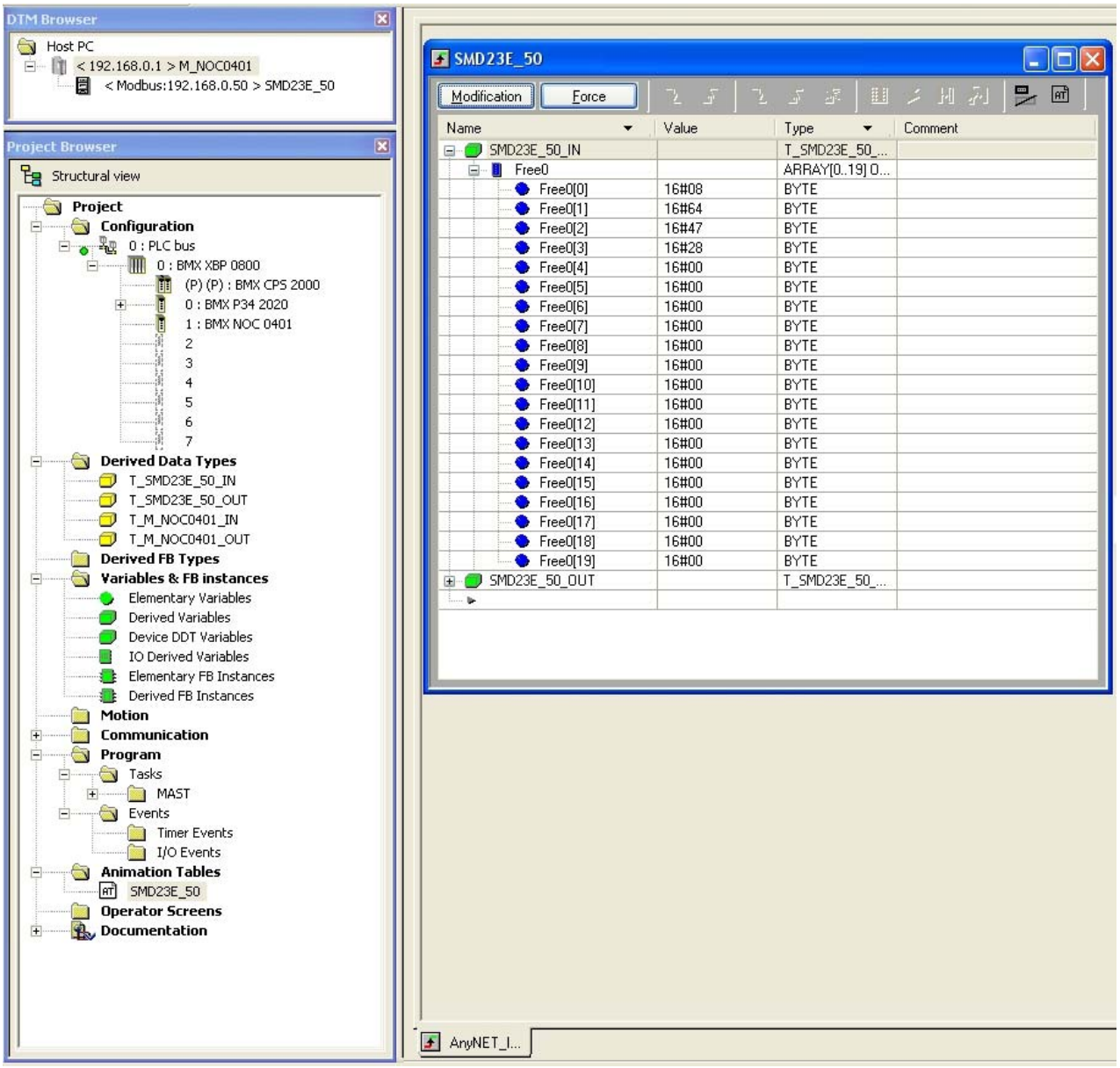


Figure 8.7 Data from an SMD23/4E

Notes

CHAPTER 9
CONFIGURATION MODE DATA FORMAT

This chapter covers the formats of the Network Output Data used to configure an SMD23/4E as well as the formats of the Network Input Data that contains the responses from the unit. An SMD23/4E requires ten 16-bit words (20 bytes) for Output Data as well as ten 16-bit words for Input Data.

Modes of Operation

An SMD23/4E has two operating modes, Configuration Mode and Command Mode. You switch between these modes by changing the state of a single bit in the Network Output Data.

Configuration Mode

Configuration Mode gives you the ability to select the proper configuration for your application without having to set any switches. SMD23/4E units are shipped from the factory without a valid configuration and must be configured on every power up. The ladder logic needed to accomplish this is included in the sample programs available from AMCI. This method simplifies change over if the unit ever needs to be replaced. A valid configuration can be saved to the unit's Flash memory and the SMD23/4E will use this as a default configuration on every power up. If you use this method, you can still write down a new configuration to the unit at any time. The new configuration is stored in RAM and is lost on power down unless you issue a command to store the new configuration in Flash.

NOTE The SMD23/4E will not accept move commands, and the motor will not receive power, until the unit has a valid configuration.

Command Mode

This mode gives you the ability to program and execute stepper moves, and reset errors when they occur. An SMD23/4E will always power up in this mode, but a valid configuration must be sent to it before the unit will apply power to the motor or allow you to issue move commands. The command data formats are described in the following chapter.

Multi-Word Format

Many of the hosts that can be used with an SMD23/4E only support 16 bit integers, which limits the range of values from -32,768 to 32,767 or 0 to 65,535. Many parameters of an SMD23/4E exceed this range. These parameters are transmitted in two separate words. The table below shows how values are split. Note that negative values are written as negative numbers in both words.

Value	First Word	Second Word
12	0	12
12,345	12	345
1,234,567	1,234	567
-7,654,321	-7,654	-321

Table 9.1 Multi-Word Format Examples

NOTE When using the Modbus-TCP protocol, use "signed 16-bit integer" as the data type.

Output Data Format

The correct format for the Network Output Data when the SMD23/4E is in Configuration Mode is shown below. EtherNet/IP and Modbus TCP addresses are both shown.

EtherNet/IP Word	Modbus TCP Register	Configuration Data	Range
0	1024	Control Word	See below
1	1025	Configuration Word	See below
2	1026	Starting Speed: Upper Word	Combined value between 1 and 1,999,999 steps/sec.
3	1027	Starting Speed: Lower Word	
4	1028	Motor Steps/Turn	200 to 32,767
5	1029	Reserved	Set to zero
6	1030	Encoder Pulses/Turn	Set to 1,024
7	1031	Idle Current Percentage	0 to 100%
8	1032	Motor Current (X10)	10 to 34, Represents 1.0 to 3.4 Arms
9	1033	Reserved	Set to zero

Table 9.2 Network Output Data Format: Configuration Mode

Control Word Format

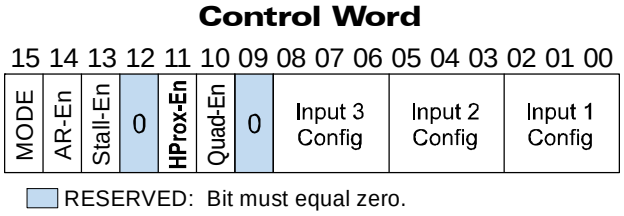


Figure 9.1 Configuration Mode: Control Word Format

Bit 15: Mode Bit – "1" for Configuration Mode Programming, "0" for Command Mode Programming. An SMD23/4E powers up in Command Mode and shows a configuration error, (hexadecimal value of 6408h), unless a valid configuration has been written to the flash memory of the SMD23/4E. The SMD23/4E will not power the motor or accept commands until a valid configuration is written to it.

Bit 14: Anti-Resonance Enable Bit – "0" enables the anti-resonance feature of the SMD23/4E. "1" disables the anti-resonance feature. The Anti-resonance feature will provide smoother operation in most cases. If you are still experiencing resonance problems with this feature enabled, disable this feature and test the machine again.

Bit 13: Stall Detection Enable Bit – "0" disables motor stall detection. "1" enables motor stall detection. Only valid with the SMD23/4E/E units that have a built-in encoder. The encoder pulses/turn must be programmed to a value of 1,024 and the encoder must be enabled by setting bit 10 to "1". (See Bit 10 below.)

Bit 11: Home Proximity Bit Enable Bit – "0" when Home Proximity bit is not used when homing the SMD23/4E. "1" when the Home Proximity bit is used when homing the SMD23/4E. Note that this bit is not the Home Proximity Bit, but enables or disables its operation. Do not use the Home Proximity bit if you only want to home to the Home Limit Switch. (Leave this bit equal to "0".)

Bit 10: Quadrature Encoder Enable Bit – "0" when Quadrature Encoder is not used. "1" to enable a Quadrature Encoder. Only valid with the SMD23/4E/E units that have the built-in encoder.

Configuration Mode - Output Data Format (continued)

Control Word Format (continued)

These bits are available on all SMD23/4E units, but Input 3 is not available with the SMD23/4E-M12 units.

Bits 8-6: Input 3 Configuration Bits – See Table Below

Bits 5-3: Input 2 Configuration Bits – See Table Below

Bits 2-0: Input 1 Configuration Bits – See Table Below

Bits				
8	7	6		
5	4	3		
2	1	0	Function	Available On
0	0	0	General Purpose Input	The input is not used in any of the functions of the SMD23/4E, but it's status is reported in the Network Data. This allows the input to be used as a discrete DC input to the host controller.
0	0	1	CW Limit	Input defines the mechanical end point for CW motion.
0	1	0	CCW Limit	Input defines the mechanical end point for CCW motion.
0	1	1	Start Indexed Move	Starts the move that is currently located in the output registers.
0	1	1	Start Indexed Move / Capture Encoder Value	When the encoder is enabled on an SMD23/4E/E, the encoder position value is captured whenever this input transitions. An inactive-to-active state transition will also trigger an Indexed Move if one is pending in the SMD23/4E/E.
1	0	0	Stop Jog or Registration Move	Brings a Jog or Registration Move to a controlled stop.
1	0	0	Stop Jog or Registration Move & Capture Encoder Value	When the encoder is enabled on an SMD23/4E/E, the encoder position value is captured when the input triggers a controlled stop to a Manual or Registration move.
1	0	1	Emergency Stop	All motion is immediately stopped when this input makes an inactive-to-active transition.
1	1	0	Home	Used to define the home position of the machine.
1	1	1	Invalid Combination	This bit combination is reserved.

Table 9.3 Configuration Mode: Control Word Format

NOTE Input 3 is not available with the SMD23/4E-M12 units. On these units, this input should be programmed to be a general purpose input by setting bits 6, 7, and 8 to "0".

Configuration Mode - Output Data Format (continued)

Configuration Word Format

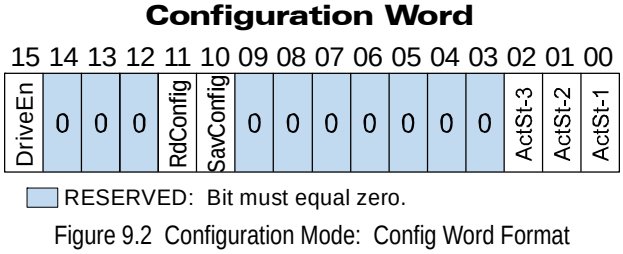


Figure 9.2 Configuration Mode: Config Word Format

Bit 15: Driver Enable Bit – "0" to disable the motor driver circuitry. "1" to enable the motor driver circuitry. When the motor driver is disabled, all voltage is removed from the motor.

Bits 14 - 12: Reserved – Must equal zero.

Bit 11: Read Present Configuration – If this bit is set when you enter Configuration Mode, the SMD23/4E responds by placing the present configuration data in the Network Input Data. You cannot write new configuration data to the unit while this bit is set. The format of the Configuration Data is given in the ***Input Data Format*** section of this chapter, starting on page 89.

Bit 10: Save Configuration – An SMD23/4E will store its configuration data to flash memory when this bit makes a 0 → 1 transition. The validity of the configuration data is checked before being written to the flash memory. If the data is not correct, the transition on this bit is ignored. Once the write to flash is completed, the SMD23/4E will write 0xAAAA into the last status word if the write was successful and the Status LED will begin flashing green, or the SMD23/4E will write 0xEEEE into the last status word if the write was unsuccessful and the Status LED will begin flashing red. Once the SMD23/4E issues its response to the Save Configuration command, it stops responding to commands and you must cycle power to the unit. This design decision prevents the SMD23/4E from responding to constant save commands from the host controller. The flash memory has a minimum of 10,000 write cycles.

Bits 9 - 3: Reserved – Must equal zero.

Bit 2: Input 3 Active Level Bit – Determines the active state of Input 3. Set to "0" if your sensor has Normally Closed (NC) contacts and the input is active when there is no current flow through it. Set to "1" if your sensor has Normally Open (NO) contacts and current flows through the input when it is active.

NOTE This input is not available with the SMD23/4E-M12 units. On these units, this input should be programmed to be "Normally Closed" by setting this bit to "0". This effectively disables the input on these units.

Bit 1: Input 2 Active Level Bit – Determines the active state of Input 2. Set to "0" if your sensor has Normally Closed (NC) contacts and the input is active when there is no current flow through it. Set to "1" if your sensor has Normally Open (NO) contacts and current flows through the input when it is active.

Bit 0: Input 1 Active Level Bit – Determines the active state of Input 1. Set to "0" if your sensor has Normally Closed (NC) contacts and the input is active when there is no current flow through it. Set to "1" if your sensor has Normally Open (NO) contacts and current flows through the input when it is active.

Configuration Mode - Output Data Format (continued)

Notes on Other Configuration Words

- Information on the ***Multi-Word Format*** used when programming the Starting Speed can be found on page 85.
- Changes to the Idle Current only take effect at the *end of the first move after re-configuration.*

Input Data Format

The format for the Network Input Data when an SMD23/4E is in Configuration Mode is shown below. EtherNet/IP and Modbus TCP addresses are both shown

EtherNet/IP Word	Modbus TCP Register	Configuration Data
0	0	Control Word
1	1	Mirror of Output Data Config Word
2	2	Mirror of Starting Speed: Upper Word
3	3	Mirror of Starting Speed: Lower Word
4	4	Mirror of Motor Steps/Turn
5	5	0000
6	6	Mirror of Encoder Pulses/Turn
7	7	Mirror of Idle Current Percentage
8	8	Mirror of Motor Current (X10)
9	9	0000 or Status message when writing Configuration data to flash memory.

Table 9.4 Network Input Data Format: Configuration Mode

Control Word Format (Word 0)

When the Configuration data is valid and accepted, this word mirrors the value of the Control Word written to the SMD23/4E. When the SMD23/4E is not configured, or the data written to it is invalid, then this word has the same format of the Status Bits MSW word when the SMD23/4E is in Command Mode. This format is explained in the ***Status Bits MSW Format*** section starting on page 104. On power up, the value of this word will be 6408h unless a valid configuration had been previously written to the flash memory of the unit.

Invalid Configurations

The following configurations are invalid:

1) Setting any of the reserved bits in the configuration words.
2) Setting any parameter to a value outside of its valid range. This includes setting the Lower Word of the Starting Speed to a value greater than 999.
3) You configure two or more inputs to have the same function, such as two CW Limit Switches.
4) Setting the *Stall Detection Enable Bit* without configuring the SMD23/4E/E to use the encoder.
5) Setting the Input Configuration bits for any input to “111”. See table 9.3 on page 87 for more information.

CHAPTER 10
COMMAND MODE DATA FORMAT

This chapter covers the formats of the Network Output Data used to command the SMD23/4E as well as the formats of the Network Input Data that contains the responses from the unit. An SMD23/4E requires ten 16-bit words (20 bytes) for Output Data as well as ten 16-bit words for Input Data.

Multi-Word Format

Many of the hosts that can be used with the SMD23/4E only support 16 bit integers, which limits the range of values from -32,768 to 32,767 or 0 to 65,535. Many parameters of the SMD23/4E exceed this range. These parameters are transmitted in two separate words. The table below shows how values are split. Note that negative values are written as negative numbers in both words. Modbus-TCP users that are limited to sixteen bit integers should used signed 16-bit integer values.

Value	First Word	Second Word
12	0	12
1,234,567	1,234	567
-7,654,321	-7,654	-321

Figure 10.1 Multi-Word Format Examples

Command Bits Must Transition

Commands are only accepted when the command bit makes a 0→1 transition. The easiest way to do this is to write a value of zero into the Command Bits MSW before writing the next command.

This condition also applies when switching from Configuration Mode to Command Mode. If a bit is set in the Control Word (Word 0) while in Configuration Mode and you switch to Command Mode with the same bit set, the command will not occur because the bit must transition between writes to the unit.

Output Data Format

The following table shows the format of the output network data words when writing command data to the SMD23/4E. EtherNet/IP and Modbus TCP addresses are both shown.

EtherNet/IP Word	Modbus TCP Register	Function
0	1024	Command Bits MSW
1	1025	Command Bits LSW
2	1026	Command Parameters Word meaning depends on the command set to the SMD23/4E
3	1027	
4	1028	
5	1029	
6	1030	
7	1031	
8	1032	
9	1033	

Table 10.1 Command Mode Data Format

Command Bits MSW

Command Bits MSW

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
MODE	PrstEnc	RunAMv	PrgAMS	PrgAMv	ClrErr	PrstPos	–JogMv	+JogMv	–Home	+Home	I-Stop	ResmMv	HoldMv	RelMv	AbsMv

Figure 10.2 Command Bits MSW Format

Bit 15: Mode Bit – "1" for Configuration Mode Programming "0" for Command Mode Programming. The SMD23/4E powers up in Command Mode and shows a configuration error, (hexadecimal value of 6408h), if a valid configuration has not been stored in flash memory. An SMD23/4E will not power the motor or accept commands until a valid configuration is written to it or read from flash memory.

Bit 14: Preset Encoder Bit – Valid only with an SMD23/4E/E, when set to "1" the SMD23/4E/E will preset the Encoder Position to the value stored in Output Words 2 and 3.

Bit 13: Run Assembled Move – When set to "1" an SMD23/4E will run the Assembled Move already stored in memory.

- **Assembled Move Type – Command Bits LSW, Bit 9:** This bit determines the type of move that is run. When this bit equals "0", a Blend Move is run. When this bit equals "1", a Dwell Move is run. When starting a Dwell Move, the Dwell Time is programmed in word 9 of the Command Data. The value is programmed in milliseconds and can range from 0 to 65,536.
- **Blend Move Direction – Command Bits LSW, Bit 4:** This bit is used to determine the direction that the Blend Move will be run in. When this bit equals "0", the Blend Move runs in the clockwise direction. When this bit equals "1", the Blend Move is run in the counter-clockwise direction.

Bits 11 & 12: Program Move & Program Move Segment Bits – These bits are used to program the segments of an Assembled Move before the move can be run. Their use is explained in the ***Assembled Move Programming*** section of this manual starting on page 36.

Bit 10: Reset Errors – When set to "1" the SMD23/4E will clear all existing errors and attempt to use the present data to run a new command. This command also resets the *Move Complete* bit.

Bit 9: Preset Motor Position Bit – When set to "1" the SMD23/4E will preset the Motor Position to the value stored in Output Words 2 and 3. This command also resets the *Move Complete* bit.

Bit 8: – Jog Move (CCW) – When set to "1" the SMD23/4E will run a Jog Move in the counter-clockwise direction. A full explanation of a ***±Jog Move*** can be found starting on page 29.

- **Run Registration Move – Command Bits LSW, Bit 7:** When this bit equals "0", and a Jog Move command is issued, it will run as a standard Jog Move. When this bit equals "1" and a Jog Move command is issued, the move will run as a Registration Move.

Bit 7: + Jog Move (CW) – When set to "1" the SMD23/4E will run a Jog Move in the clockwise direction. A full explanation of a ***±Jog Move*** can be found starting on page 29.

- **Run Registration Move – Command Bits LSW, Bit 7:** When this bit equals "0", and a Jog Move command is issued, it will run as a standard Jog Move. When this bit equals "1" and a Jog Move command is issued, the move will run as a Registration Move.

Command Bits MSW (continued)

Bit 6: **Find Home CCW –** When set to "1" the SMD23/4E will attempt to move to the Home Limit Switch in the counter-clockwise direction. A full explanation of homing can be found in the ***Homing The SMD23/4E*** chapter starting on page 41.

Bit 5: **Find Home CW –** When set to "1" the SMD23/4E will attempt to move to the Home Limit Switch in the clockwise direction. A full explanation of homing can be found in the ***Homing The SMD23/4E*** chapter starting on page 41.

Bit 4: **Immediate Stop Bit –** When set to "1" the SMD23/4E will stop all motion without deceleration. The Motor Position value will become invalid if this bit is set during a move. Setting this bit when a move is not in progress will not cause the Motor Position to become invalid.

Bit 3: **Resume Move Bit –** Set to "1" to resume a move that you previously placed in a hold state. Use of the Resume Move and Hold Move bits can be found in the ***Controlling Moves In Progress*** section of this manual starting on page 38. Note that a move in its hold state does not need to be resumed. The move is automatically cancelled if another move is started in its place.

Bit 2: **Hold Move Bit –** Set to "1" to hold a move. The move will decelerate to its programmed Starting Speed and stop. The move can be completed by using the Resume Move bit or it can be aborted by simply starting another move. Use of the Hold Move and Resume Move bits can be found in the ***Controlling Moves In Progress*** section of this manual starting on page 38.

Bit 1: **Relative Move Bit –** Set to "1" to perform a Relative Move using the data in the rest of the Command Data. The full explanation of a ***Relative Move*** can be found starting on page 27.

Bit 0: **Absolute Move Bit –** Set to "1" to perform an Absolute Move using the data in the rest of the Command Data. The full explanation of an ***Absolute Move*** can be found starting on page 28.

Command Bits LSW

Command Bits LSW

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
DriveEn	0	0	0	BkHPrx	0	AMType	IndxMv	RunReg	0	SavAMv	AMvDir	HyCtrl	EncMv	0	0

Figure 10.3 Command Bits LSW Format

Bit 15: **Driver Enable Bit –** "0" to disable the motor current, "1" to enable motor current. A valid configuration must be written to the SMD23/4E before the driver can be enabled.

Bits 14 – 12: **Reserved –** Must equal "0".

Bit 11: **Backplane Home Proximity Bit –** When the SMD23/4E is configured to use the Backplane Home Proximity bit, the unit will ignore the state of the Home Input as long as this bit equals "0". This bit must equal "1" before a transition on the Home Input can be used to home the machine. Further information on using the Home Proximity bit can be found in the ***Homing Inputs*** section starting on page 41.

Command Bits LSW (continued)

Bit 10: **Reserved –** Must equal "0".

Bit 9: **Assembled Move Type –** When this bit equals "0", a Blend Move is started when the Run Assembled Move bit, (Command Word 1, Bit 13) makes a 0 → 1 transition. When this bit equals "1", a Dwell Move is started on the transition. The direction of a Blend Move is controlled by the Blend Move Direction bit, (Command Bits LSW, Bit 4). In a Dwell Move, the Dwell Time between segments is programmed in Word 9 of the command data.

Bit 8: **Run As An Indexed Move –** If this bit is set when a move command is issued, the SMD23/4E will not run the move immediately, but will instead wait for an inactive-to-active transition on an input configured as a *Start Indexer Move* input. The move command data, including this bit, must remain in the Network Output Data registers while performing an Indexed Move.

Bit 7: **Run Registration Move –** When this bit equals "0", and a Jog Move command is issued, it will run as a standard Jog Move. When this bit equals "1" and a Jog Move command is issued, the move will run as a Registration Move.

Bit 6: **Reserved –** Must equal "0".

Bit 5: **Save Assembled Move –** Set this bit to save the programmed Assembled Move to flash memory. This bit must be set when the Move Program Mode makes a 1 → 0 transition at the end of the programming cycle to write the data to flash memory. The SMD23/4E responds by flashing the Status LED when the writing is complete. If the LED is flashing green, the write to flash memory was successful. If it flashes red, then there was an error in writing the data. In either case, power must be cycled to the SMD23/4E before you can continue. This design decision is to protect the flash memory from constant write commands. The flash memory has a minimum of 10,000 write cycles.

Bit 4: **Blend Move Direction –** When you command a Blend Move to run, this bit determines the direction of rotation. Set to "0" for a clockwise Blend Move, '1' for a counter-clockwise Blend Move.

Bit 3: **Reserved –** Must equal "0".

Bit 2: **Encoder Move –** Valid with an SMD23/4E/E only, set this bit to "1" when you command an Absolute or Relative move to cause the move distance to be based on the encoder position, not on the motor position. Set to "0" for a standard Relative or Absolute move based on motor position. Please note that an encoder move does not stop at the programmed encoder position. It starts to decelerate at this position. A full description of ***Encoder Moves*** starts on page 32.

Bit 1: **Reserved –** Must equal "0".

Bit 0: **Reserved –** Must equal "0".

Command Blocks

The following section lists the output data format for the sixteen different commands. EtherNet/IP and Modbus TCP addresses are both shown.

Absolute Move

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0001
1	1025	***Command Bits LSW***		See pg. 93
2	1026	Abs. Target Position: Upper Word	Steps	Combined value between –8,388,607 and +8,388,607
3	1027	Abs. Target Position: Lower Word		
4	1028	Programmed Speed: Upper Word	Steps/Second	Combined value between the Configured Starting Speed and 2,999,999
5	1029	Programmed Speed: Lower Word		
6	1030	Acceleration	Steps/ms/sec	1 to 5000
7	1031	Deceleration	Steps/ms/sec	1 to 5000
8	1032	*Reserved*		Must equal zero for compatibility with future releases.
9	1033	Acceleration Jerk		0 to 5000

Table 10.2 Absolute Move Command Block

Relative Move

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0002
1	1025	***Command Bits LSW***		See pg. 93
2	1026	Rel. Target Position: Upper Word	Steps	Combined value between –8,388,607 and +8,388,607
3	1027	Rel. Target Position: Lower Word		
4	1028	Programmed Speed: Upper Word	Steps/Second	Combined value between the Configured Starting Speed and 2,999,999
5	1029	Programmed Speed: Lower Word		
6	1030	Acceleration	Steps/ms/sec	1 to 5000
7	1031	Deceleration	Steps/ms/sec	1 to 5000
8	1032	*Reserved*		Must equal zero for compatibility with future releases.
9	1033	Acceleration Jerk		0 to 5000

Table 10.3 Relative Move Command Block

Command Blocks (continued)

Hold Move

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0004
1	1025	***Command Bits LSW***		See pg. 93
2	1026	Unused		See Note Below
3	1027	Unused		See Note Below
4	1028	Unused		See Note Below
5	1029	Unused		See Note Below
6	1030	Unused		See Note Below
7	1031	Unused		See Note Below
8	1032	Unused		See Note Below
9	1033	Unused		See Note Below

Table 10.4 Hold Move Command Block

Unused words are ignored by an SMD23/4E and can be any value, including parameter values in the previous command.

Resume Move

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0008
1	1025	***Command Bits LSW***		See pg. 93
2	1026	Unused		See Note Below
3	1027	Unused		See Note Below
4	1028	Programmed Speed: Upper Word	Steps/Second	Combined value between the Configured Starting Speed and 2,999,999
5	1029	Programmed Speed: Lower Word		
6	1030	Acceleration	Steps/ms/sec	1 to 5000
7	1031	Deceleration	Steps/ms/sec	1 to 5000
8	1032	*Reserved*		Must equal zero for compatibility with future releases.
9	1033	Acceleration Jerk		0 to 5000

Table 10.5 Resume Move Command Block

Unused words are ignored by the SMD23/4E and can be any value, including parameter values in the previous command. This is typically the case when resuming a move, the words are listed as “Unused” to highlight that the target position of a held move cannot be changed when the move is resumed.

Command Blocks (continued)

Immediate Stop

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0010
1	1025	***Command Bits LSW***		See pg. 93
2	1026	Unused		See Note Below
3	1027	Unused		See Note Below
4	1028	Unused		See Note Below
5	1029	Unused		See Note Below
6	1030	Unused		See Note Below
7	1031	Unused		See Note Below
8	1032	Unused		See Note Below
9	1033	Unused		See Note Below

Table 10.6 Immediate Stop Command Block

Unused words are ignored by the SMD23/4E and can be any value, including parameter values in the previous command.

+Find Home (CW)

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0020
1	1025	***Command Bits LSW***		See pg. 93
2	1026	Unused		See Note Below
3	1027	Unused		See Note Below
4	1028	Programmed Speed: Upper Word	Steps/Second	Combined value between the Configured Starting Speed and 2,999,999
5	1029	Programmed Speed: Lower Word		
6	1030	Acceleration	Steps/ms/sec	1 to 5000
7	1031	Deceleration	Steps/ms/sec	1 to 5000
8	1032	*Reserved*		Must equal zero for compatibility with future releases.
9	1033	Acceleration Jerk		0 to 5000

Table 10.7 +Find Home (CW) Command Block

Unused words are ignored by the SMD23/4E and can be any value, including parameter values in the previous command.

Command Blocks (continued)

–Find Home (CCW)

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0040
1	1025	***Command Bits LSW***		See pg. 93
2	1026	Unused		See Note Below
3	1027	Unused		See Note Below
4	1028	Programmed Speed: Upper Word	Steps/Second	Combined value between the Configured Starting Speed and 2,999,999
5	1029	Programmed Speed: Lower Word		
6	1030	Acceleration	Steps/ms/sec	1 to 5000
7	1031	Deceleration	Steps/ms/sec	1 to 5000
8	1032	*Reserved*		Must equal zero for compatibility with future releases.
9	1033	Acceleration Jerk		0 to 5000

Table 10.8 –Find Home (CCW) Command Block

Unused words are ignored by the SMD23/4E and can be any value, including parameter values in the previous command.

+Jog Move (CW)

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0080
1	1025	***Command Bits LSW***		See pg. 93 Bits 7 & 6 must equal “00”
2	1026	Unused		See Note Below
3	1027	Unused		See Note Below
4	1028	Programmed Speed: Upper Word	Steps/Second	Combined value between the Configured Starting Speed and 2,999,999
5	1029	Programmed Speed: Lower Word		
6	1030	Acceleration	Steps/ms/sec	1 to 5000
7	1031	Deceleration	Steps/ms/sec	1 to 5000
8	1032	*Reserved*		Must equal zero for compatibility with future releases.
9	1033	Acceleration Jerk		0 to 5000

Table 10.9 +Jog Move (CW) Command Block

Unused words are ignored by the SMD23/4E and can be any value, including parameter values in the previous command.

Command Blocks (continued)

+Registration Move (CW)

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0080
1	1025	***Command Bits LSW***		See pg. 93 Bits 7 & 6 must equal "10"
2	1026	Stopping Distance: Upper Word	Steps	Combined value between 0 and +8,388,607
3	1027	Stopping Distance: Lower Word		
4	1028	Programmed Speed: Upper Word	Steps per Second	Combined value between the Configured Starting Speed and 2,999,999
5	1029	Programmed Speed: Lower Word		
6	1030	Acceleration	Steps/ms/sec	1 to 5000
7	1031	Deceleration	Steps/ms/sec	1 to 5000
8	1032	Min. Reg. Move Distance: Upper Word	Steps	Combined value between 0 and +8,388,607
9	1033	Min. Reg. Move Distance: Lower Word		

Table 10.10 +Registration Move (CW) Command Block

–Jog Move (CCW)

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0100
1	1025	***Command Bits LSW***		See pg. 93 Bits 7 & 6 must equal "00"
2	1026	Unused		See Note Below
3	1027	Unused		See Note Below
4	1028	Programmed Speed: Upper Word	Steps/Second	Combined value between the Configured Starting Speed and 2,999,999
5	1029	Programmed Speed: Lower Word		
6	1030	Acceleration	Steps/ms/sec	1 to 5000
7	1031	Deceleration	Steps/ms/sec	1 to 5000
8	1032	*Reserved*		Must equal zero for compatibility with future releases.
9	1033	Acceleration Jerk		0 to 5000

Table 10.11 –Jog Move (CCW) Command Block

Unused words are ignored by the SMD23/4E and can be any value, including parameter values in the previous command.

Command Blocks (continued)

–Registration Move (CCW)

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0100
1	1025	***Command Bits LSW***		See pg. 93 Bits 7 & 6 must equal "10"
2	1026	Stopping Distance: Upper Word	Steps	Combined value between 0 and +8,388,607
3	1027	Stopping Distance: Lower Word		
4	1028	Programmed Speed: Upper Word	Steps per Second	Combined value between the Configured Starting Speed and 2,999,999
5	1029	Programmed Speed: Lower Word		
6	1030	Acceleration	Steps/ms/sec	1 to 5000
7	1031	Deceleration	Steps/ms/sec	1 to 5000
8	1032	Min. Reg. Move Distance: Upper Word	Steps	Combined value between 0 and +8,388,607
9	1033	Min. Reg. Move Distance: Lower Word		

Table 10.12 –Registration Move (CCW) Command Block

Preset Position

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0200
1	1025	***Command Bits LSW***		See pg. 93
2	1026	Position Preset Value: Upper Word	Steps	Combined value between –8,388,607 and +8,388,607
3	1027	Position Preset Value: Lower Word		
4	1028	Unused		See Note Below
5	1029	Unused		See Note Below
6	1030	Unused		See Note Below
7	1031	Unused		See Note Below
8	1032	Unused		See Note Below
9	1033	Unused		See Note Below

Table 10.13 Preset Position Command Block

Unused words are ignored by the SMD23/4E and can be any value, including parameter values in the previous command.

➤ Presetting the position will also reset the *Move Complete* status bit.

Command Blocks (continued)

Reset Errors

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0400
1	1025	***Command Bits LSW***		See pg. 93
2	1026	Unused		See Note Below
3	1027	Unused		See Note Below
4	1028	Unused		See Note Below
5	1029	Unused		See Note Below
6	1030	Unused		See Note Below
7	1031	Unused		See Note Below
8	1032	Unused		See Note Below
9	1033	Unused		See Note Below

Table 10.14 Reset Errors Command Block

Unused words are ignored by the SMD23/4E and can be any value, including parameter values in the previous command.

➤ Resetting errors will also reset the *Move Complete* status bit.

Run Assembled Move

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#2000
1	1025	***Command Bits LSW***		See pg. 93 Blend Move: Bit 9 = "0" Dwell Move: Bit 9 = "1" Blend Move direction set by Bit 4.
2	1026	Unused		See Note Below
3	1027	Unused		See Note Below
4	1028	Unused		See Note Below
5	1029	Unused		See Note Below
6	1030	Unused		See Note Below
7	1031	Unused		See Note Below
8	1032	Unused		See Note Below
9	1033	Unused with Blend Move Dwell Time with Dwell Move	milliseconds	0 to 65,535

Table 10.15 Run Assembled Move Command Block

Unused words are ignored by the SMD23/4E and can be any value, including parameter values in the previous command.

Command Blocks (continued)

Preset Encoder Position

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#4000
1	1025	***Command Bits LSW***		See pg. 93
2	1026	Encoder Preset Value: Upper Word	Steps	Combined value between –8,388,607 and +8,388,607
3	1027	Encoder Preset Value: Lower Word		
4	1028	Unused		See Note Below
5	1029	Unused		See Note Below
6	1030	Unused		See Note Below
7	1031	Unused		See Note Below
8	1032	Unused		See Note Below
9	1033	Unused		See Note Below

Table 10.16 Preset Encoder Position Command Block

Unused words are ignored by the SMD23/4E and can be any value, including parameter values in the previous command.

Programming Blocks

The following blocks are used to program an Assembled Move. Both of the moves, Blend Move, and Dwell Move, are programmed exactly the same way. The bit configuration used when starting the move determines which type of Assembled Move is run.

First Block

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#0800
1	1025	***Command Bits LSW***		See pg. 93
2 - 9	1026 - 1033	Unused		See Note Below

Table 10.17 Assembled Move First Programming Block

Unused words are ignored by the SMD23/4E and can be any value, including parameter values from the previous command.

Once the first block is transmitted, the SMD23/4E responds by setting bits 8 and 9 in the Status Bits MSW word. (See ***Status Bits MSW Format*** starting on page 104.) Once these are set, you can then start transmitting Segment Blocks.

Segment Block

EtherNet/IP Word	Modbus TCP Register	Function	Units	Range
0	1024	***Command Bits MSW***		16#1800
1	1025	***Command Bits LSW***		See pg. 93
2	1026	Rel. Target Position: Upper Word	Steps	Combined value between –8,388,607 and +8,388,607
3	1027	Rel. Target Position: Lower Word		
4	1028	Programmed Speed: Upper Word	Steps/Second	Combined value between the Configured Starting Speed and 2,999,999
5	1029	Programmed Speed: Lower Word		
6	1030	Acceleration	Steps/ms/sec	1 to 5000
7	1031	Deceleration	Steps/ms/sec	1 to 5000
8	1032	*Reserved*		Must equal zero for compatibility with future releases.
9	1033	Acceleration Jerk		0 to 5000

Table 10.18 Assembled Move Segment Programming Block

Note that each Segment Block starts with bits 11 and 12 set in the Command Bits MSW word (16#1800). When the SMD23/4E sees bit 12 of Command Bits MSW set, it will accept the block and reset bit 9 in the Status Bits MSW word. When your program sees this bit reset, it must respond by resetting bit 12 of Command Bits MSW. The SMD23/4E will respond to this by setting bit 9 in the Status Bits MSW word and the next Segment Block can be written to the SMD23/4E. You can write a maximum of sixteen Segment Blocks for each Assembled Move.

Input Data Format

The correct format for the Network Input Data when an SMD23/4E is in Command Mode is shown below. EtherNet/IP and Modbus TCP addresses are both shown.

EtherNet/IP Word	Modbus TCP Register	Command Mode Input Data
0	0	Status Bits MSW
1	1	Status Bits LSW
2	2	Motor Position: Upper Word
3	3	Motor Position: Lower Word
4	4	Encoder Position: Upper Word
5	5	Encoder Position: Lower Word
6	6	Captured Encoder Position: Upper Word
7	7	Captured Encoder Position: Lower Word
8	8	Value of Motor Current (X10)
9	9	Value of Acceleration Jerk Parameter

Table 10.19 Network Input Data Format: Command Mode

Status Bits MSW Format

Status Bits MSW

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
MODE	CrtlOK	ConErr	CmdErr	InErr	PosInvld	Xmt_MS	PgmMov	MvCmp	Decel	Accel	Homed	Stopped	Hold	MvCCW	MvCW

Figure 10.4 Command Mode: Status Bits MSW Format

Bit 15: Mode Bit – "1" for Configuration Mode Programming, "0" for Command Mode Programming. An SMD23/4E will power up in Command Mode and shows a configuration error unless a valid configuration was saved in flash memory. An SMD23/4E will not power the motor or accept commands until a valid configuration is written to it or read from its internal flash memory.

Bit 14: Controller OK Bit – "1" when the SMD23/4E is operating without a fault, "0" when an internal fault condition exists.

Bit 13: Configuration Error Bit – "1" on power up before a valid configuration has been written to the SMD23/4E or after any invalid configuration has been written to the driver. "0" when the SMD23/4E has a valid configuration written to it or if valid configuration data is read from flash memory on power up.

Bit 12: Command Error Bit – "1" when an invalid command has been written to the SMD23/4E. This bit can only be reset by the Reset Errors bit, Command Bits MSW, Bit 10.

Bit 11: Input Error Bit – "1" when:

- ➤Emergency Stop input has been activated.
- ➤Either of the End Limit Switches activates during any move operation except for homing.
- ➤Starting a Jog Move in the same direction as an active End Limit Switch.
- ➤If the opposite End Limit Switch is reached during a homing operation.

This bit is reset by a ***Reset Errors*** command. The format of the command is given on page 101.

Input Data Format (continued)

Status Bits MSW Format (continued)

Bit 10: Position Invalid Bit – "1" when:

- ➤A configuration is written to the SMD23/4E
- ➤The motor position has not been preset
- ➤The machine has not been homed
- ➤The Network Connection has been lost and re-established.
- ➤An Immediate or Emergency Stop has occurred
- ➤An End Limit Switch has been reached
- ➤A motor stall has been detected.

Absolute moves cannot be performed while the position is invalid.

Bit 9: Transmit Move Segment Bit – The SMD23/4E sets this bit to tell the host that it is ready to accept the data for the next segment of your assembled move profile. Its use is explained in the ***Assembled Move Programming*** section of this manual starting on page 36.

Bit 8: Program Move Mode Bit – The SMD23/4E sets this bit to signal the host that it is ready to accept assembled move profile programming data. Its use is explained in the ***Assembled Move Programming*** section of this manual starting on page 36.

Bit 7: Move Complete Bit – Set to "1" when the present Absolute, Relative, Jog, Registration, or Assembled Move command completes without error. This bit is reset to "0" when the next move command is written to the SMD23/4E, when the position is preset, or a Reset Errors command is issued to the unit. This bit is also set along with the Command Error bit (Bit 12 of this word), when any Jog Move or Registration Move parameters are outside of their valid ranges. This bit is not set on a command error for any other type of command. Finally, this bit is not set at the end of a homing operation.

Bit 6: Decelerating Bit – Set to "1" when the present move is decelerating. Set to "0" at all other times.

Bit 5: Accelerating Bit – Set to "1" when the present move is accelerating. Set to "0" at all other times.

Bit 4: Homing Complete Bit – Set to "1" when a homing command has completed successfully, "0" at all other times.

Bit 3: Axis Stopped Bit – Set to "1" when the motor is not in motion. Note that this is stopped for any reason, not just a completed move. For example, an Immediate Stop command during a move will set this bit to "1", but the Move Complete Bit, (bit 7 above) will not be set.

Bit 2: Hold State Bit – Set to "1" when a move command has been successfully brought into a Hold State. Hold States are explained is the Controlling Moves In Progress section starting on page 22.

Bit 1: CCW Move Bit – Set to "1" when the motor is rotating in a counter-clockwise direction.

Bit 0: CW Move Bit – Set to "1" when the motor is rotating in a clockwise direction.

Input Data Format (continued)

Status Bits LSW Format

Status Bits LSW

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
DrvEn	Stall	0	0	HeartBt	LmtCon	InvPar	MtnLag	DrvFlt	NetLoss	PLCState	TempWarn	0	IN3St	IN2St	IN1St

Figure 10.5 Command Mode: Status Bits LSW Format

Bit 15: Driver Enabled Bit – Present state of global Driver Enable Bit. When set to "0", current is not available to the motor under any condition. When set to "1", the motor driver section of the SMD23/4E is enabled and current is available to the motor, but current may be removed for other reasons. Motor current is removed if there is a Driver Fault (Bit 7 below), or if the motor is idle and Idle Current Reduction is programmed to its *To 0%* setting. Note that the motor will still receive power under an E-Stop condition.

Bit 14: Motor Stall Detected Bit – Set to "1" when a motor stall has been detected. This is only available on SMD23/4E_E units that have been configured to detect stalls. This bit will always equal "0" on SMD23/4E units that do not have a built-in encoder.

Bit 13: Reserved Bit – Will always equal zero.

Bit 12: Reserved Bit – Will always equal zero.

Bit 11: Heart Beat Bit – This bit will change state approximately every 500 milliseconds. Monitor this bit to verify that the unit and network connection are operating correctly. Note that this bit is only available when in Command Mode.

Bit 10: Limit Condition Active Bit – This bit is set if an End Limit Switch is reached during a move. This bit will be reset when the Limit Switch changes from its active to inactive state, or when a Reset Errors Command is issued.

Bit 9: Invalid Parameter Change Bit – Set during a Jog Move if parameters are changed to invalid values. Thew only way to clear this bit once set is to write correct parameter values to the SMD23/4E. Parameters that can be changed during a Jog Move are Programmed Speed, Acceleration, and Deceleration.

Bit 8: Reserved Bit – Will always equal zero.

Bit 7: Driver Fault Bit – If the driver section of the SMD23/4E is enabled, this bit will be a "1" during an Overtemperature Fault. This bit can only be cleared by cycling power to the SMD23/4E.

Bit 6: Network Lost Error Bit – If the physical network connection is lost at any time, this bit will be set when the connection is re-established. The Input Error bit will also be set. Note that this bit is not set if the communication loss is not due to a physical error.

Bit 5: PLC State Bit – On ControlLogix and CompactLogix platforms, this bit equals "1" when the PLC is in Program mode and "0" when in Run mode. This bit will always equal "0" on all other platforms.

Bit 4: Temperature Warning Bit – This bit is set to "1" when the processor internal temperature exceeds 90°C. At this point, the heatsink temperature is typically near 83°C. If this bit trips often and you want to lower the operating temperature of the unit, consider installing a fan to force additional airflow around the unit.

Bit 3: Reserved Bit – Will always equal zero.

Input Data Format (continued)

Status Bits LSW Format (continued)

Bit 2: **Input3 State Bit –** "1" when Input 3 is in its active state. The active state of the input is programmed as explained in the ***Configuration Word Format*** section starting on page 88.

NOTE This input is not available with the SMD23/4E-M12 units. On these units, this input should be programmed to be Normally Closed by setting its Input Active Level Bit to "0" in the configuration data. This effectively disables the input on these units. See bit 2 of the ***Configuration Word Format*** on page 88 for more information.

Bit 1: **Input2 State Bit –** "1" when Input 2 is in its active state. The active state of the input is programmed as explained in the ***Configuration Word Format*** section starting on page 88.

Bit 0: **Input1 State Bit –** "1" when Input 1 is in its active state. The active state of the input is programmed as explained in the ***Configuration Word Format*** section starting on page 88.

Motor Position Data

These two inputs words report the motor position based on the number of steps output by the SMD23/4E. The range for the position value is -32,768,000 to 32,767,999. The data is transmitted in the ***Multi-Word Format*** described on page 91. Note that both words in the multi-word format will be negative if the position value is negative.

Encoder Position Data

These two inputs words report the encoder position based on the quadrature pulses from the internal encoder of an SMD23/4E/E. The range for the encoder position value is -32,768,000 to 32,767,999. The data is transmitted in the ***Multi-Word Format*** described on page 91. Note that both words in the multi-word format will be negative if the encoder position value is negative.

Captured Encoder Position Data

These two inputs words report the position of the internal encoder of an SMD23/4E/E when an input transitions for its inactive to its active state. The input must be configured as a Capture Encoder Value input or as a Stop Jog and Capture Encoder Value input. The range for the captured encoder position value is -32,768,000 to 32,767,999. The data is transmitted in the Multi-Word Format described on page 51. Note that both words in the multi-word format will be negative if the captured encoder position value is negative.

Notes on Clearing a Driver Fault

A Driver Fault occurs when there is an over temperature condition. When a Driver Fault occurs, the SMD23/4E sets bit 7 of the Status Bits LSW word in the Network Input Data. The only way to clear this fault is to lower the temperature of the motor and cycle power to the SMD23/4E.

Notes

APPENDIX A

IP ADDRESS SETUP WITH BOOTP

It is possible to use a BOOTP server to set the IP address of an SMD23/4E. This appendix explains how to use the BOOTP server from Rockwell Automation to set the IP address.

Initial Configuration

Starting at the beginning of chapter 6, ***Network Configuration***, follow those instructions up to the point where you have your network interface card configured to communicate with the SMD23/4E. This is explained in the section ***Configure Your Network Interface Card (NIC)*** which is on page 60.

Using the RA BOOTP Server

Attach the SMD23/4E

The first step in configuring the SMD23/4E when using a BOOTP server is attaching the module to your computer. Any Cat5, 5e, or 6 cable can be used. You can attach the SMD23/4E directly to your computer or use a network switch. The SMD23/4E has an "auto-switch" port which eliminates the need for a crossover cable in direct connect applications.

NOTE **Do not power up the SMD23/4E at this point. Just attach it to your network.**

Start Your BOOTP Server

If needed, start your BOOTP server. The BOOTP-DHCP server software, version 2.3, from Rockwell Automation is used in this example. The R.A. BOOTP server window is broken down into two panes, "Request History" and "Relation List". "Request History" tells you what responses come over the network and the "Relation List" shows the setup data you have entered.

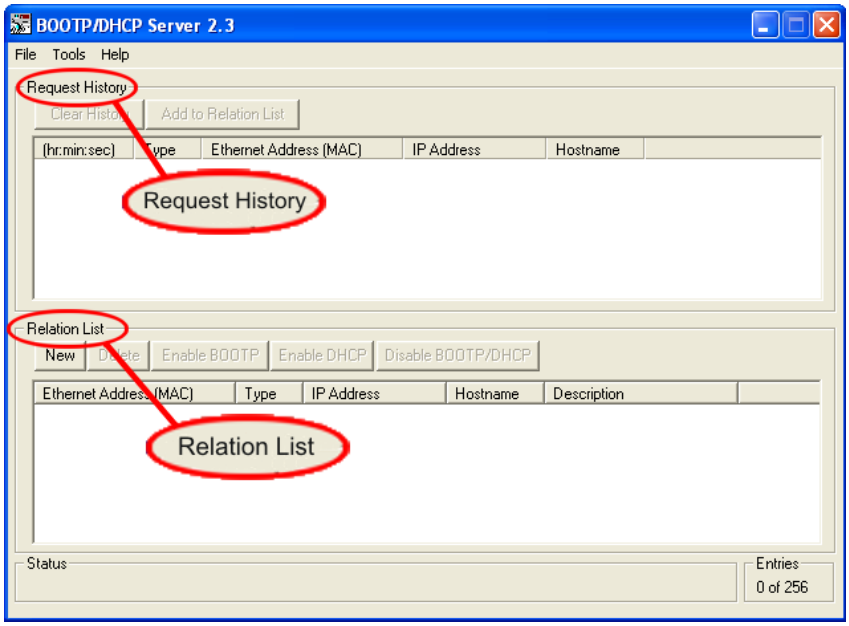


Figure A.1 RA BOOTP Server

Using the RA BOOTP Server (continued)

Enabling BOOTP Protocol

Changing the IP address of the SMD23/4E requires you to enable the BOOTP protocol on the unit before you can change the IP address. The SMD23/4E has the BOOTP protocol disabled by default.

1) In the "Relation List" pane of the RA BOOTP Server software, click on [New]. In the window that opens, enter the MAC address of the SMD23/4E which is printed on the module's serial number tag. You do not have to enter the "-" characters when entering the address on the screen. You must enter the IP address that the SMD23/4E presently has. This is 192.168.0.50 by default. The hostname and Description fields can be left blank. Click [OK].

2) Apply power to the SMD23/4E and wait for the Module LED to come on solid green and the Network LED to become flashing green. Both of these LED's are on the rear cover.

3) Click on your new entry in the "Relation List". This will activate the buttons in the pane. Click on the [Enable BOOTP] button. The message "[Enable BOOTP] Command successful" should appear instantly in the status line at the bottom of the window. Figure A.2 shows the state of the RA BOOTP Server software at this point.

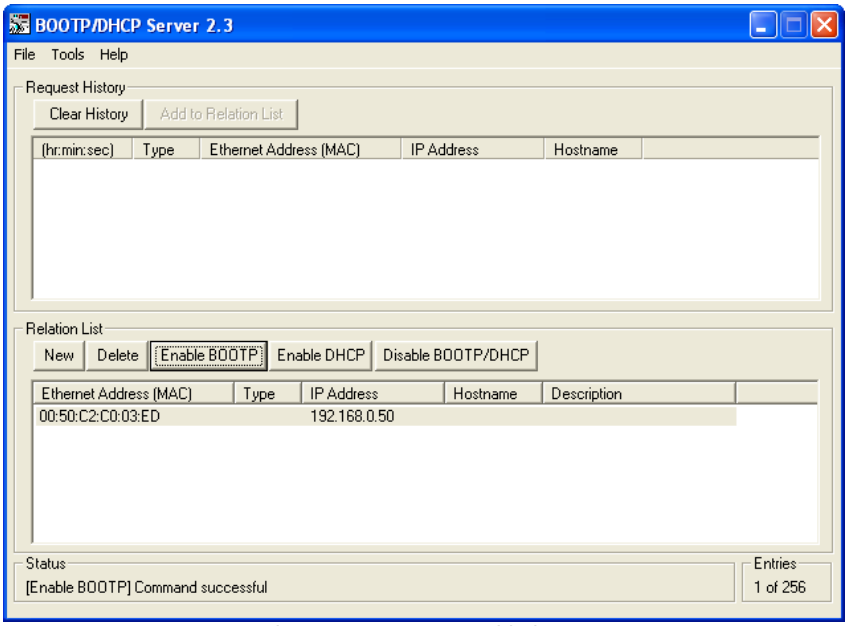


Figure A.2 BOOTP Enabled

4) The BOOTP protocol is now enabled on the SMD23/4E.

NOTE **Remove power from the unit before continuing.**

Using the RA BOOTP Server (continued)

Setting the IP Address

With the BOOTP protocol enabled on the SMD23/4E, you can now change the IP address of the unit.

1) Double click on your new entry in the "Relation List" This will bring up the Properties window again. Enter the new IP address for the SMD23/4E and click [OK].

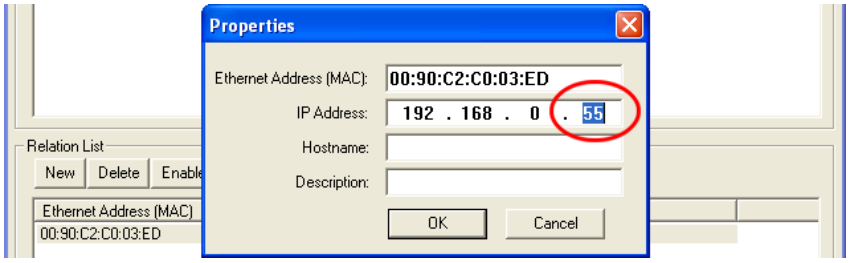


Figure A.3 Defining the New IP Address

2) Click on *Tools -> Network Setting*. In the window that opens, set the subnet mask and the default gateway.

NOTE The *Default Gateway* must be set to a valid value, even if the SMD23E will never need to communicate with a different subnet. This means that the default gateway must be on the same subnet as the SMD23/4E but cannot be the address used by the SMD23/4E. AMCI suggests setting the default gateway to the address of your host controller.

3) Apply power to the module and wait for the Unit Status LED to come on solid green and the Network Status LED to be flashing green. At this point, you should also have a message in the "Request History" pane that lists the MAC address of the SMD23/4E along with the IP address you requested.

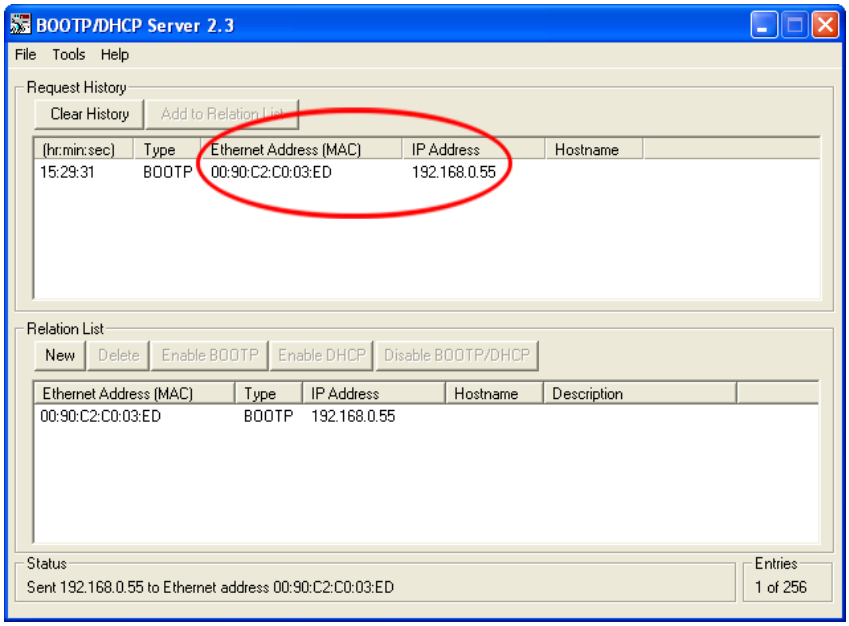


Figure A.4 New IP Address Set in SMD23/4E

Using the RA BOOTP Server (continued)

Disabling the BOOTP Protocol

Unless your target system has a BOOTP server running all of the time, you must disable the BOOTP protocol on the SMD23/4E. In order to comply with ODVA specifications, the SMD23E will issue BOOTP requests on power up until a valid response is received. There is no time out value. If BOOTP is enabled on the SMD23E and a BOOTP server is not available, the SMD23E will not be able to join the network.

To disable BOOTP protocol on the SMD23/4E:

1) With power still applied to the unit, click on your new entry in the "Relation List". This will enable the buttons above it.
2) Click on the [Disable BOOTP/DHCP] button. The message "[Disable BOOTP] Command successful" should appear instantly in the status line at the bottom of the window.

The SMD23/4E is now configured.

Testing the New IP Address

The easiest way to test the new address of the SMD23/4E is with the "ping" command. Before you can use the command, you have to be sure the SMD23/4E and your computer are still on the same subnet. For example, if the new address of the SMD23/4E is 192.168.0.42 and your computer has and address of 192.168.0.1, with a subnet mask of 255.255.255.0, then the two pieces of equipment are on the same subnet. (In this case, the first three numbers of the IP address must match.) If the new address of the SMD23/4E is 192.168.50.50, then the computer and SMD23/4E are not on the same subnet and you must go back into the Network Configuration panel and change your adapter's TCP/IP settings. For more information, refer back to the ***Configure Your Network Interface Card (NIC)*** section of this manual, starting on page 60.

Once you are sure your computer and SMD23/4E are on the same subnet, open the DOS terminal if necessary by clicking on the [Start] button, and clicking on [Run...]. A dialog box will open. Enter 'cmd' on the text line and press [Enter] on the keyboard.

Once the terminal is open, type in 'ping aaa.bbb.ccc.ddd' where 'aaa.bbb.ccc.ddd' in the new IP address of the SMD23/4E. The computer will ping the SMD23/4E and the message "Reply from aaa.bbb.ccc.ddd: bytes=32 time<10ms TTL=128" should appear four times.

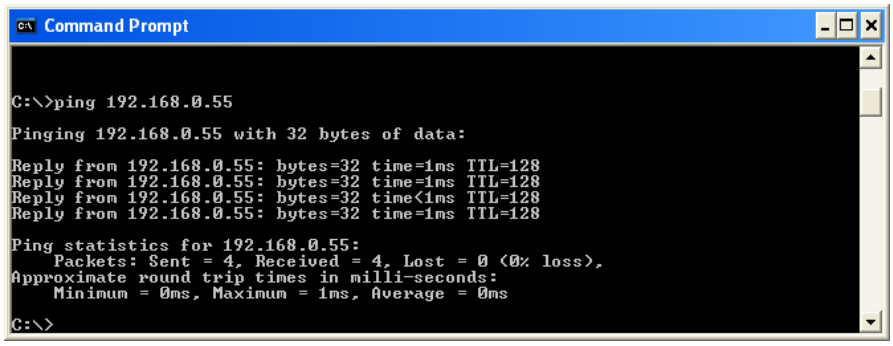

Command Prompt
C:\>ping 192.168.0.55
Pinging 192.168.0.55 with 32 bytes of data:
Reply from 192.168.0.55: bytes=32 time=1ms TTL=128
Reply from 192.168.0.55: bytes=32 time=1ms TTL=128
Reply from 192.168.0.55: bytes=32 time<1ms TTL=128
Reply from 192.168.0.55: bytes=32 time=1ms TTL=128
Ping statistics for 192.168.0.55:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>

Figure A.5 IP Address Changed Successfully

If the message "Request timed out." or "Destination host unreachable" appears, then one of three things has occurred:

- You did not enter the correct address in the ping command.
- The new IP address of the SMD23/4E was not set correctly.
- The SMD23/4E and the computer are not on the same subnet.

Notes

ADVANCED MICRO CONTROLS INC.

20 GEAR DRIVE, TERRYVILLE, CT 06786 T: (860) 585-1254 F: (860) 584-1973

www.amci.com

LEADERS IN ADVANCED CONTROL PRODUCTS